OPTIMIZATION OF TRAFFIC SIGNAL SETTINGS IN NETWORKS BY MIXED-INTEGER LINEAR PROGRAMMING

Nathan Cartner, et al

Massachusetts Institute of Technology

Prepared for:

Army Research Office-Durham

March 1974

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Report No. 91	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER AD-779446
4. TITLE (and Subtitle) OPTIMIZATION OF TRAFFIC SIGNAL SETTINGS IN NETWORKS BY MIXED-INTEGER LINEAR PROGRAMMING		5. TYPE OF REPORT & PERIOD COVERED Technical Report March 1974 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Nathan Gartner John D.C. Little Henry Gabbay		8. CONTRACT OR GRANT NUMBER(s) DAHC04-73-C0032
9. PERFORMING ORGANIZATION NAME AND ADDRESS M.I.T. Operations Research Center 77 Massachusetts Avenue, Room 24-215 Cambridge, MA 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS P-9239-M
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE March 1974 13. NUMBER OF PAGES 133
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) U.S. Army Research Office-Durham Box CM, Duke Station Durham, NC 27706		15. SECURITY CLASS. (of this report) Unclassified 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Releasable without limitations on dissemination.

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Traffic Signals
Mixed-Integer Linear Programming
Synchronization
Minimal Delay

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A mixed-integer linear programming formulation is developed for minimizing delay to traffic in a signal controlled road network. Offsets, splits of green time and a common cycle time for the network are considered as decision variables simultaneously. The traffic flow pattern is modeled as a periodic platoon, and a link performance function is derived in the form of a piecewise linear convex surface representing the delay incurred by these platoons. Stochastic effects are accounted for by a saturation deterrence function re-

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

Unclassified
SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20. presenting the expected overflow queue on each link and are included as an additive component in the objective function. Computational results, using the MPSX system, are given for an arterial with 11 signals in Waltham, Mass., and a portion of the UTCS network in Washington, D.C. containing 20 nodes, 63 links and 21 loops. (U)

OPTIMIZATION OF TRAFFIC SIGNAL SETTINGS IN NETWORKS BY MIXED-INTEGER LINEAR PROGRAMMING

BY

NATHAN GARTNER

JOHN D.C. LITTLE

HENRY GABBAY

Technical Report No. 91

Work Performed Under

Subcontract From

KLD Associates, Huntington, N.Y.

(DOT Contract FH-11-7924)

Variable Cycle Signal Timing Program

M.I.T. OSP 80687

and

Army Research Office

Contract DAHC04-73-C0032

Methods of Operations Research

P-9239-M M.I.T. OSP 81219

Operations Research Center

Massachusetts Institute of Technology

Cambridge, Massachusetts 02139

March 1974

ic

FOREWORD

The Operations Research Center at the Massachusetts Institute of Technology is an interdepartmental activity devoted to graduate education and research in the field of operations research. The work of the Center is supported, in part, by government contracts and industrial grants-in-aid. The work reported herein was supported (in part) by KLD Associates, Inc., Huntington, N.Y., in connection with Department of Transportation Contract FH-11-7924 and (in part) by the Army Research Office under Contract DAHC04-73-C0032.

John D.C. Little
Director

ABSTRACT

A mixed-integer linear programming formulation is developed for minimizing delay to traffic in a signal controlled road network. Offsets, splits of green time and a common cycle time for the network are considered as decision variables simultaneously. The traffic flow pattern is modeled as a periodic platoon, and a link performance function is derived in the form of a piecewise linear convex surface representing the delay incurred by these platoons. Stochastic effects are accounted for by a saturation deterrence function representing the expected overflow queue on each link and are included as an additive component in the objective function. Computational results, using the MPSX system, are given for an arterial with 11 signals in Waltham, Massachusetts, and a portion of the UTCS network in Washington, D.C. containing 20 nodes, 63 links and 21 loops.

Table of Contents

List of Tables

List of Figures

1. INTRODUCTION

The last fifteen years have seen a gradual infusion of computer technology and operations research methods into traffic signal systems. This has manifested itself in three main ways:

- in offline methods for calculating signal settings
- in "solid-state" digital control hardware, and
- in real-time closed-loop control systems using both on-line traffic detection and calculation of settings.

We shall describe each briefly so as to relate our work to the field in general.

Offline analytic methods may be considered to start with the works of Clayton[5] and Webster [40] in Britain, for determining green splits and cycle time at an isolated intersection. For coordinated signals modern research begins with Morgan and Little's rigorous maximization of bandwidth on an artery[28]. Little[24] subsequently generalized and extended the methods to networks. SIGOP [37] introduced a quadratic link performance function and a new, but not global, method of optimization. Hillier [15] and Allsop [1] developed the Combination Method. This permits the use of an arbitrary link performance function and produces global optimal offsets for fixed values of green splits and cycle time. Gartner[8] formulated the problem in terms of dynamic programming optimization and developed an efficient network partitioning algorithm. A different direction was taken by Robertson in TRANSYT [34]. Whereas the link performance function approach assumes that events on a link are, to a good approximation, independent of upstream offsets, TRANSYT carries more memory of previous traffic history. However, TRANSYT's optimization procedure is not global and is computationally rather slow, thereby limiting the number of variables that can reasonably be searched over. Finally, we note that Gazis[12] proposed to use a piecewise linear link performance function so that linear programming can be applied to calculate optimal offsets and that Eisner[6] adopted a simple "two-piece linear" function proposed by Newell[33] and formulated the network coordination problem in terms of mixed-integer linear programming. However, no computations have been attempted by these two workers.

On the hardware side, the new technology has produced flexible, reliable signal switching equipment. A complex network can rather easily by controlled from a central location with a small computer. Such a computer can also store a variety of pre-calculated signal settings and put them into effect by time of day or, to a limited degree, in response to inputs from vehicle detectors. The new systems have the important characteristic of being easy to change [36]. A new timing plan can be introduced by reading in a few cards at the central location.

Real-time, closed-loop systems that employ extensive vehicle detection and powerful central computers have caught the imagination of many traffic engineers and city managers. Such systems offer the hope of making the best use of a severely limited resource, namely, the existing city streets. A variety of projects have been undertaken, starting with the pioneer work in Toronto [14]. A number of worthwhile benefits have been obtained. However, such systems currently have several drawbacks, including high initial cost, considerable ongoing reliability and maintenance problems (particularly with the detectors) and, most important to our discussion, a lack of adequate theory for deciding how to go in some optimal manner from detector input to signal settings.

The present work is part of a research and development program undertaken by the Federal Highway Administration in connection with the UTCS (Urban Traffic Control System) in Washington, D.C. The program seeks to proceed from simple systems to complex ones, through a series of "generations." Briefly, first generation consists of fixed-time settings calculated offline and stored in simple signal coordination hardware. Second generation also uses fixed-time settings but calculates them online in response to traffic detection. Third generation embodies continuous computer-calculated control.

The research approach described in the following pages has been first to develop a traffic model and performance measure and then to bring contemporary optimization techniques to bear on the problem. The principal technique used is mixed-integer linear programming and so the method has been given the acronym MITROP (mixed integer traffic optimization program). In the researchers' view the results obtained so far are both significant and promising. The emphasis on a systematic approach has brought new advances applicable to first and, potentially, second and third generation systems.

The main features of MITROP are:

- the simultaneous consideration of all major control variables, i.e., offsets, splits, and cycle time
- a global optimization technique
- the use of link performance functions, and
- a minimum delay objective function that includes not only the usual deterministic queue buildup and discharge but also the stochastic effects of overflow queues. The latter occur as flow rates approach intersection capacity.

The characteristics of MITROP can be brought out by comparing it to the Combination Method (CM) and TRANSYT. Consider CM first. Both CM and MITROP use link performance functions but MITROP permits simultaneous optimization over splits and cycle time. CM could be extended to handle cycle time also at a very considerable increase in computation time (see Gartner and Little [11] but the inclusion of splits is virtually out of the question (see Korsak [23]). TRANSYT contains a more detailed representation of traffic than any of the systems mentioned above. However, it is not yet clear how significant this is at moderate flows since TRANSYT and SIGOP have been found comparable in field evaluations both in Glasgow and San Jose [41 , 22] . Furthermore, TRANSYT uses one variable at a time search for estimating optimal values for the decision variables. Convergence is to a local, not global optimum and therefore the results depend on the starting settings. Computation is relatively slow and although, in principle, all control variables (splits, offsets, and cycle time) can be examined, in practice, search has been confined to offsets, primarily.

MITROP is the first method to search over all control variables: offsets, splits, and cylce time with a globally optimal method. The most useful part of this increased scope appears to be the consideration of cycle time. Sensitivity testing on sample networks indicate significant improvements in performance resulting from letting cycle time vary. For example the optimal settings for a particular test network analyzed in the report when all variables are considered simultaneously shows an improvement of 11.6% in overall performance relative to settings obtained when the control variables are considered separately.

It is important to understand the physical tradeoffs involved in determining cycle times. If it were not for switching losses, very short cycle times would be preferable. This is because, if capacity is adequate, delay is never more than one red period and decreases with cycle length. However, the need for increased capacity favors long cycle times. In order to make the appropriate tradeoff on a network-wide basis, it is necessary to know at every intersection how much worse the traffic situation is when cycle time is shortened. What happens on the street is that when flow is too close to capacity, a few cars will, from time to time, fail to clear the intersection during a single cycle and thereby create an overflow queue along with substantial extra delay. If flow and capacity become even closer, overflow queues happen more frequently and delay becomes worse. MITROP contains a representation of this phenommenon, via the saturation deterrence function, and thus is able to make the tradeoffs essential to calculating optimal cycle times. Recommended directions for future research, exploiting the MITROP results, are suggested in Section 8.

2. THE TRAFFIC SIGNAL NETWORK TIMING PROBLEM

This chapter presents the traffic signal network timing problem in a general form. The traffic network is modeled in terms of nodes, representing the signalized intersections, and links, representing sections of streets carrying traffic in one direction between two intersections (see Fig. 2.1). The variables and parameters of the system are defined and the physical constraining relations existing among them are stated. The timing problem is then formulated as a nonlinear optimization program involving the minimization of a generalized disutility function associated with traveling through the signalized intersections.

2.1 Nomenclature

The traffic network and the principal control variables are defined as follows:

S_i = traffic signal at node i

$N = \{i\}$ = the set of all nodes in the network

(i,j) = the link going from node i to j

$L = \{(i,j)\}$ = the set of all links in the network

$\mathcal{G} = [N,L]$ = the network graph

R_{ij} = physical red time at S_j facing link (i,j)

r_{ij} = effective red time at S_j facing (i,j)

G_{ij} = physical green time at S_j facing (i,j)

g_{ij} = effective green time at S_j facing (i,j)

A_{ij} = amber time at S_j facing (i,j)

l_{ij} = phase lost time at S_j in direction (i,j)

a_j = periods of all-red time at S_j

C = the network cycle time

All time variables defined above are measured in seconds.

The relations between physical and effective signal timings are illustrated in Fig. 2.2, following the basic model for traffic signal operation used by Webster [40] and others [5,27,39].

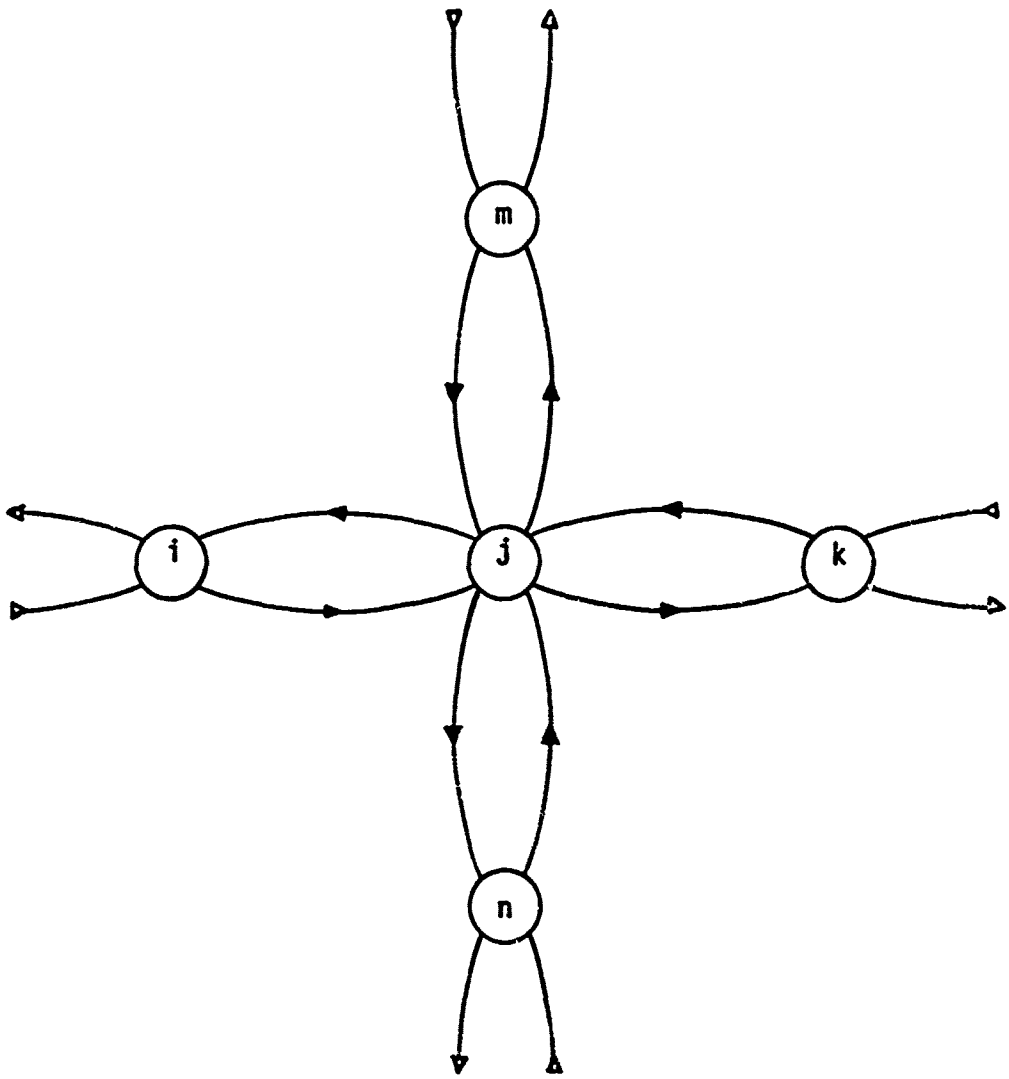
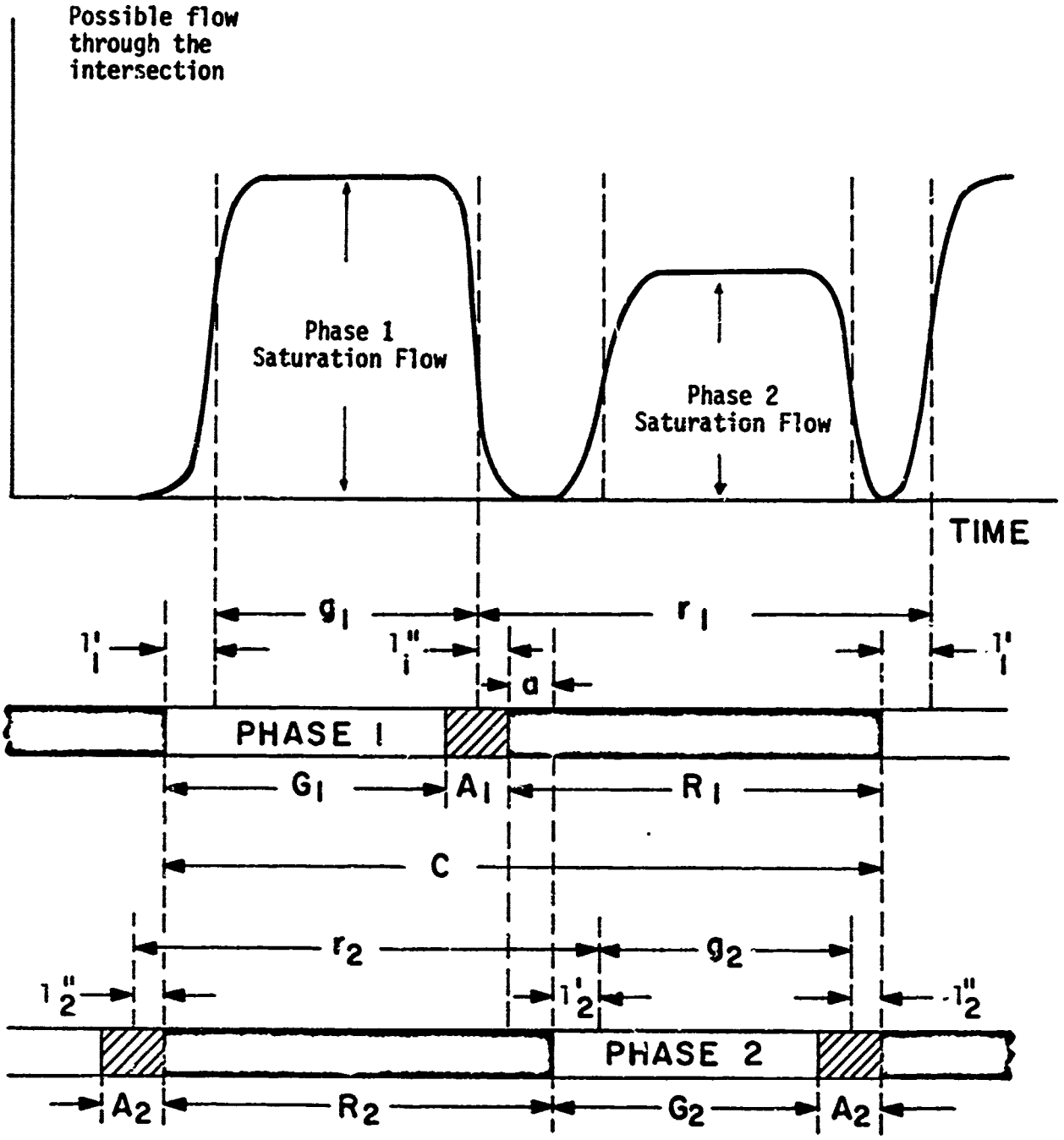


Figure 2.1: Nodes and links in a signal-controlled traffic network.



$$G + A + R = C$$

$$G + A = g + l$$

$$r + g = C$$

$$l = l' + l''$$

$$r_1 - (l_1 + a) = g_2 + l_2$$

$$r_2 - (l_2 + a) = g_1 + l_1$$

Figure 2.2 - Basic model for traffic signal operation.

Referring to Fig. 2.3 we shall now relate the concepts of offset and arrival time on a link. Vehicle platoons released during the green time of S_i travel towards S_j. Let

τ_{ij} = travel time of the platoon's leading edge from S_i to S_j on the connecting link

γ_{ij} = arrival time of the platoon's leading edge at S_j's stop line, measured relative to start of g_{ij} so that for every cycle we have,

$$- r_{ij} < \gamma_{ij} \leq g_{ij} \tag{2.1}$$

ϕ_{ij} = offset time from beginning of green at S_i to beginning of respective green at S_j so that,

$$\tau_{ij} - \gamma_{ij} = \phi_{ij} \tag{2.2}$$

θ_{ij} = offset time between S_i and S_j measured from beginning of green at S_i to beginning of next green at S_j. Consequently, we have,

$$\theta_{ij} = [\phi_{ij}]_{\text{mod } C}$$

$$0 \leq \theta_{ij} < C \tag{2.3}$$

Finally, let

$z_{ij}(\phi_{ij}, r_{ij}, g_{ij}, C)$ = average cost per vehicle on link (i,j) when traveling through S_j

f_{ij} = average flow on link (i,j) (vehicles/second).

2.2 Objective Function

z_{ij} is a disutility incurred by vehicles traveling through the signal-controlled network such as delay, stops, acceleration noise, etc., and therefore should be minimized. Hence, the objective function in an optimization procedure would be:

$$\min \sum_{(i,j)\in L} f_{ij}\, z_{ij}(\phi_{ij}, r_{ij}, g_{ij}, C) \tag{2.4}$$

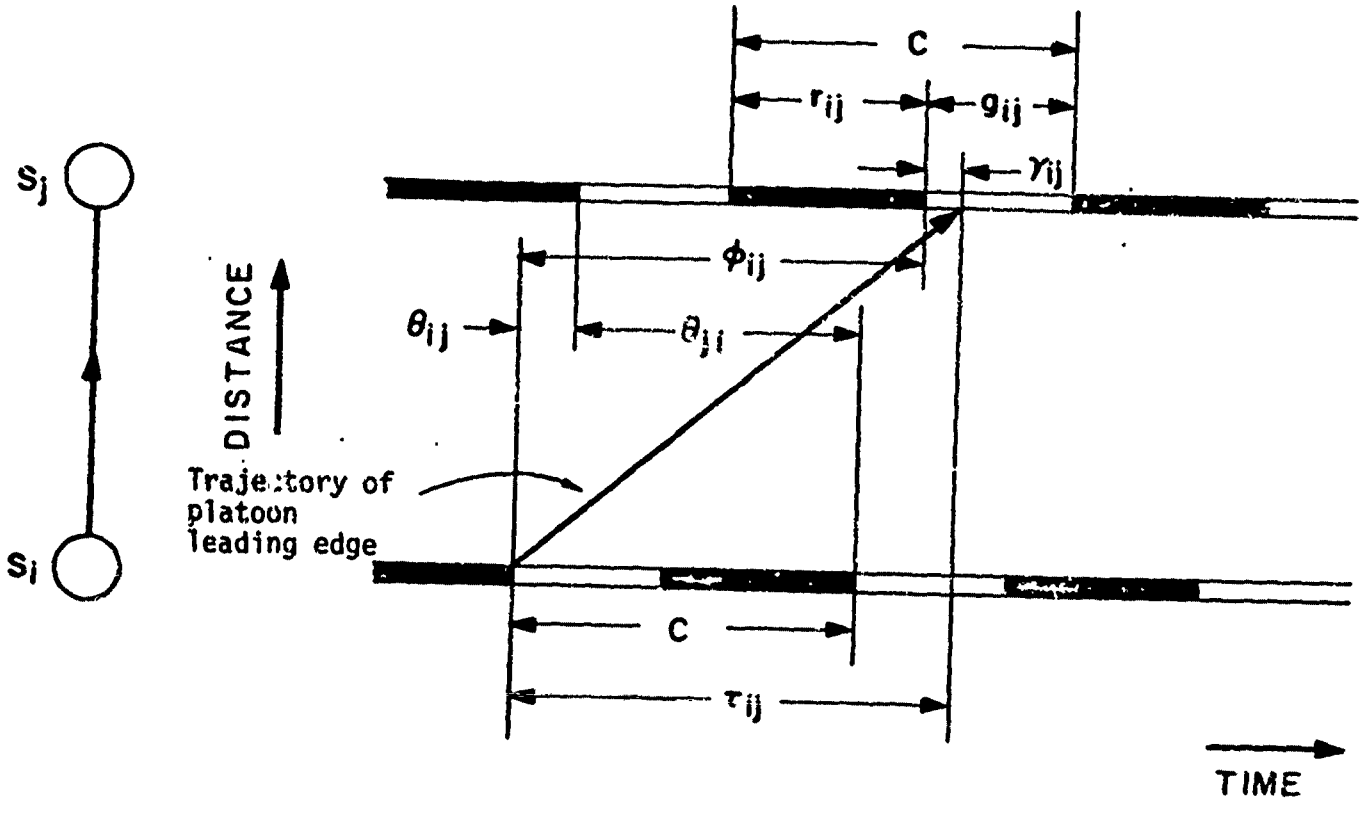


Figure 2.3 - Signal-controlled link parameters.

This formulation assumes a separability condition, i.e., the disutility on each link is a function of parameters for that link only. Newell [31] has shown in a theoretical study that this condition holds for high traffic flows and when platoons formed by the traffic lights spread so much between lights that the time between the arrival of the first and last cars of a platoon at a light exceeds the green period. The validity of this assumption is also supported by several empirical studies [10,16]. Furthermore, the British Combination Method, which premises the same assumption, has shown good results when implemented in field studies [18].

The constraints which the optimization procedure must obey are described next.

2.3 Constraints

Let

ℓ = a set of links that form a loop (undirected circuit) of the network

$\mathcal{L}$ = the set of all loops of the network

We also use the following set-theoretic notation:

$\epsilon \equiv$ that belongs to

$\forall \equiv$ for every

A physical constraint of the system is that the sum of the offsets around any loop must be an integer multiple of the cycle time, i.e.,

$$\sum_{(i,j)\epsilon\ell} \phi_{ij} = n_\ell C \qquad \forall \ell \epsilon \mathcal{L} \tag{2.5}$$

where n_ℓ is an integer variable associated with loop ℓ. Another physical constraint is

$$r_{ij} + g_{ij} = C \qquad \forall (i,j) \epsilon L \tag{2.6}$$

In order for the network to be able to handle the given flow we have the capacity constraint:

$$f_{ij} s_{ij} \geq f_{ij} C \qquad \forall (i,j) \epsilon L \tag{2.7}$$

where s_{ij} is the saturation flow rate at the downstream signal of (i,j).

To facilitate pedestrian crossing, minimum red must be established

$$r_{ij} \geq r_{ij}^{min} \tag{2.8}$$

Since the gain in capacity with very long cycle times is often insignificant, we should have an upper limit on the cycle length. This is also desirable for preventing drivers from becoming extremely impatient or believing that the signals are defective.

$$C \leq C^{max} \tag{2.9}$$

From a practical point of view, including safety considerations, it is desirable also to have a lower limit on the cycle time (i.e., an upper limit on the aspect switching rate), in addition to capacity considerations,

$$C \geq C^{min} \tag{2.10}$$

For simplicity at this stage of model development, we assume there are no special turn signals, lagged greens, etc., and the intersections involve, at most, two intersecting streets. However, link omissions such as one-way streets and T intersections are accommodated. We need a notation to say that green times are the same when approaching the signal from either direction on the same street and to express the relation between red on one street (e.g., the main street) and green on the intersecting street (e.g., the cross street). For this purpose, let

$$P_j = \{i \mid (i,j) \text{ is on main street}\}$$

$$\bar{P}_j = \{i \mid (i,j) \text{ is on cross street}\}$$

The physical red and green times are $r_{ij} - l_{ij}$ and $g_{ij} + l_{ij}$, respectively (Fig. 2.2).

The equality of physical reds and greens on opposing approaches to node j on the same street is expressed by

$$\left.\begin{aligned} r_{ij} - l_{ij} &= r_{kj} - l_{kj} \\ g_{ij} + l_{ij} &= g_{kj} + l_{kj} \end{aligned}\right\} \begin{array}{l} j \in N \\ i,k \in P_j \end{array} \tag{2.11}$$

For the equality of physical red and green on a street and its cross street we refer to Fig. 2.1 and Fig. 2.2:

$$\left.\begin{aligned} r_{ij} - (l_{ij} + a_j) &= g_{mj} + l_{mj} \\ r_{mj} - (l_{mj} + a_j) &= g_{ij} + l_{ij} \end{aligned}\right\} \begin{array}{l} j \in N \\ i \in P_j \\ m \in \bar{P}_j \end{array} \tag{2.12}$$

2.4 Optimization Problem

Summarizing the characterization given above, the network signal setting problem can be stated as a mixed-integer nonlinear optimization (MINLO) program:

MINLO: Find values of ϕ_{ij}, r_{ij}, g_{ij}, C to

$$\min \sum_{(i,j)\in L} f_{ij}\, z_{ij}\,(\phi_{ij}, r_{ij}, g_{ij}, C)$$

subject to:

$$\sum_{(i,j)\in \ell} \phi_{ij} = n_\ell\, C \qquad \forall\, \ell \in \mathcal{L}$$

$$\left.\begin{aligned} r_{ij} + g_{ij} &= C \\ g_{ij}\, s_{ij} &\geq f_{ij}\, C \\ r_{ij} &\geq r_{ij}^{min} \end{aligned}\right\} \quad \forall\, (i,j) \in L$$

$$\left.\begin{aligned} r_{ij} - l_{ij} &= r_{kj} - l_{kj} \\ g_{ij} + l_{ij} &= g_{kj} + l_{kj} \end{aligned}\right\} \quad \forall\, i,k \in P_j$$

$$\left.\begin{aligned} r_{ij} - (l_{ij} + a_j)\, g_{mi} + l_{mj} \\ r_{mj} - (l_{mj} + a_j) = g_{ij} + l_{ij} \end{aligned}\right\} \begin{array}{l} \forall\, j \in N \\ \forall\, i \in P_j \\ \forall\, m \in \bar{P}_j \end{array}$$

$$C^{min} \leq C \leq C^{max}$$

$$r_{ij}, g_{ij} \geq 0$$

$$\phi_{ij}, n_\ell \text{ unrestricted in sign; } n_\ell \text{ integer.}$$

A substantial number of the variables and constraints are redundant and can be eliminated by substitution.

Among the various candidate optimization techniques for solving this program we have chosen to use mixed-integer programming (MIP). One of the chief attractions of this approach is that reasonably efficient packages for solving such problems have recently become available, especially MPSX which runs on IBM 370 equipment [21].

3. LINK PERFORMANCE FUNCTION

The disutility associated with traveling through the signalized links is described in this section in terms of a Link Performance Function (LPF). A traffic flow model is developed for this purpose and evaluated with respect to field data.

3.1 Basic Relations

In the following discussion we establish the beginning of green time at the downstream signal of the link as a reference point and assign to it zero value. Thus, the time interval [-r,g] denotes one cycle period consisting of an effective red period [-r,0] and an effective green period [0,g] (Fig. 3.1). This section deals exclusively with traffic behavior on a specific link (i,j). To simplify notation we drop subscripts.

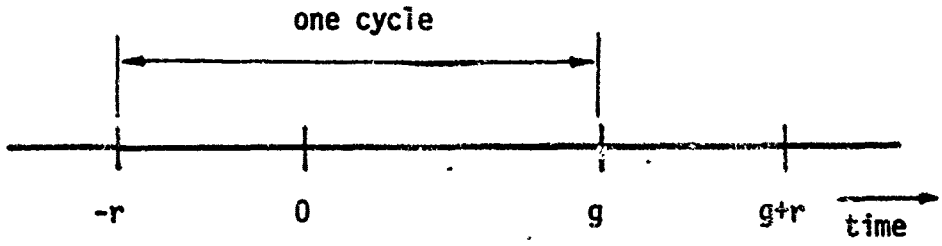


Figure 3.1: Signal parameters.

Definitions:

q(t) = arrival rate (vehicles/second)

A(t) = cumulative number of arrivals at time t (vehicles). Starting at the beginning of any red period, we have

$$A(t) = \int_{-r}^{t} q(\tau)\, d\tau \qquad (3.1)$$

s = saturation flow rate at the signal's stop line (vehicles/second)

s(t) = possible departure rate at time t. For each cycle we have:

$$s(t) = \begin{cases} 0 \text{ for } -r \leq t < 0 \\ s \text{ for } 0 \leq t < g \end{cases} \qquad (3.2)$$

The saturation flow s is maintained during the green period only as long as there is a queue waiting for service. Otherwise, the departure rate equals the arrival rate.

We first confine ourselves to deterministic flows. Stochastic effects are accounted for later in Section 5. We assume:

1. Arrivals are periodic, i.e.,

$$q(t) = q(t + n\,C) \tag{3.3}$$

where n is an integer number.

2. The signal is undersaturated, i.e.,

$$A_p < gs \tag{3.4}$$

where

$$A_p = \int_{-r}^{g} q(t)\,dt \tag{3.5}$$

is the total number of cars arriving at the signal during one cycle. gs denotes the capacity of the signal, that is, the number of possible departures during the green period (and hence also during one cycle time).

3. The arrival rate during the green period of the signal's cycle time does not exceed the saturation flow rate, i.e.,

$$q(t) \leq s \quad \text{for} \quad 0 \leq t < g. \tag{3.6}$$

(3.6) implies that once a queue has vanished during the green period it cannot rebuild before the next red period commences.

According to these assumptions, all vehicles arriving during a cycle in which the red period precedes the green can be accommodated in that cycle. It follows that the queue is always empty at the end of the green period and delay time calculations can be confined to a single interval $[-r,g]$.

The queue length $Q(t)$ at any time $t \in [-r,g]$ is given by the difference between the cumulative number of arrivals and the cumulative number of departures:

$$Q(t) = \begin{cases} A(t) & \text{if} \quad -r \leq t < 0 \\ A(t) - ts & \text{if} \quad 0 \leq t \leq t_0 \\ 0 & \text{if} \quad t_0 \leq t < g. \end{cases} \tag{3.7}$$

t_o is the queue clearance time and denotes the first time during the green period when the queue disappears. By definition $t = t_o$ when

$$Q(t) = A(t_o) - t_o\, s = 0 \text{ and } 0 \leq t_o \leq g. \tag{3.8}$$

The delay incurred by Q(t) queueing vehicles during an interval dt is Q(t) dt. Hence, the total delay time Z incurred by traffic during one cycle is represented by the area under the queue length curve:

$$Z(\gamma,r) = \int_{-r}^{g} Q(t)\, dt = \int_{-r}^{t_o} Q(t)\, dt \tag{3.9}$$

The average delay per vehicle $z(\gamma,r)$ is obtained by dividing by the total number of arrivals during one cycle:

$$z(\gamma,r) = \frac{1}{A_p}\, Z(\gamma,r) \tag{3.10}$$

Note that z is a function of arrival time and split and is independent of cycle time.

The subsequent discussion considers primarily delays as the cost to be minimized. However, the models that are developed can be used also to calculate a more general link performance function combining costs of delays, stops, acceleration noise, or other measures of effectiveness, by using appropriate weighting factors [4,19].

3.2 Traffic Flow Model

We construct a simple model to analyze the link performance function. We assume the traffic flow pattern on the link to be a rectangular-shaped periodic platoon. The platoon is characterized by its time-length p and constant flow rate q (Fig. 3.2).

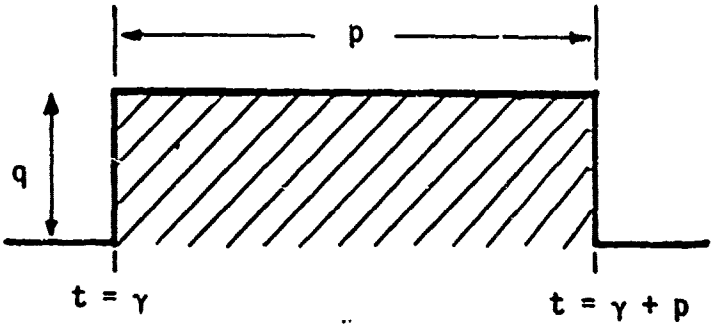


Figure 3.2: Traffic flow pattern.

We can always determine for this model a period of one cycle time in which:

$$q(t) = \begin{cases} q & \text{if } \gamma < t < \gamma + p \\ 0 & \text{else.} \end{cases} \tag{3.11}$$

The platoon size for this model is given according to (3.5) by:

$$A_p = qp \tag{3.12}$$

Referring to Fig. 2.1 we describe how to determine the traffic flow parameters on link (j,k). Given the effective green time at the upstream end of link (j,k), g_{ij}, we assume that the platoon length p_{jk} when entering link (j,k) is:

$$p_{jk} = g_{ij} \tag{3.13}$$

If the average flow on the link is f_{jk} (vehicles/second), we further assume that

$$q_{jk} = f_{jk} \frac{C}{g_{ij}} \text{ (veh/sec)}, \tag{3.14}$$

i.e., the flow is assumed to be uniformly distributed over the entire green phase. Alternative assumptions, such as a 'tadpoling' flow pattern [20], or other patterns, could be equally made.

Equations (3.13) and (3.14) refer to the primary flow on link (j,k). If there is a substantial secondary flow, the platoon length is modified as shown in Fig. 3.3 (we use now superscripts ' and " for primary and secondary flows, respectively). Consequently we have,

$$p = p' + p''. \tag{3.15}$$

p' is determined solely by the green phase serving the primary flow on link (j,k), i.e., $p'_{jk} = g_{ij}$ as in eqn. (3.13) above. Similary, q_{jk} is determined following (3.14),

$$q_{jk} = f'_{jk} \frac{C}{g_{ij}} \text{ (veh/sec)} \tag{3.16}$$

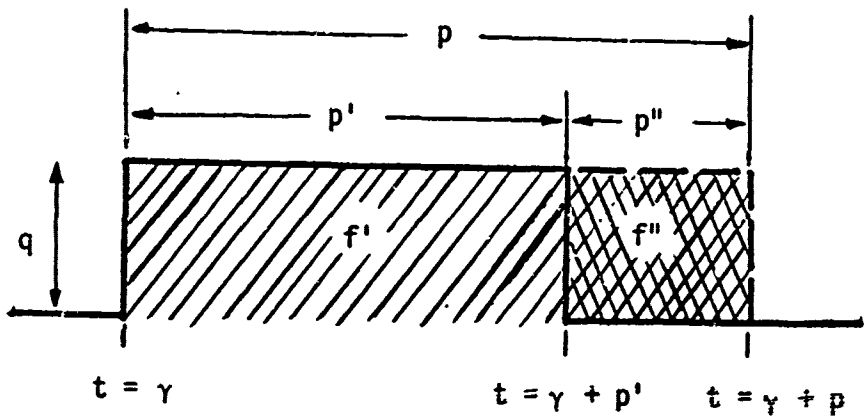


Figure 3.3: Primary and secondary flows.

In order to maintain a uniform platoon, the extended portion due to the secondary flow is calculated as follows:

$$p''_{jk} = \frac{f''_{jk}}{q_{jk}} C = \frac{f''_{jk}}{f'_{jk}} g_{ij} = \frac{f''_{jk}}{f'_{jk}} p'_{jk} \tag{3.17}$$

Hence,

$$p_{jk} = p'_{jk} \left(1 + \frac{f''_{jk}}{f'_{jk}}\right) \tag{3.18}$$

In certain cases the turning-in traffic constitutes the primary flow, for instance, the flow from (m,j) onto (j,k) in Fig. 2.1 may be the primary flow on link (j,k). In these cases p' is determined by the appropriate green time from which it emanated (g_{mj} in this example).

To determine the link performance function we need p and q at the downstream end of the link. Both parameters are a function of the link length d, owing to the dispersion phenomenon. A simple relationship, corroborated by some empirical evidence [13,30] is assumed:

$$p(d) = \begin{cases} p(0) \times k(d) & \text{if } p(d) < C \\ C & \text{if } p(d) \geq C \end{cases} \tag{3.19}$$

where k(d) is a piecewise linear relationship as shown in Fig. 3.4. The platoon length cannot exceed the length of one cycle time, at which point flow is becoming continuous (see also [42]).

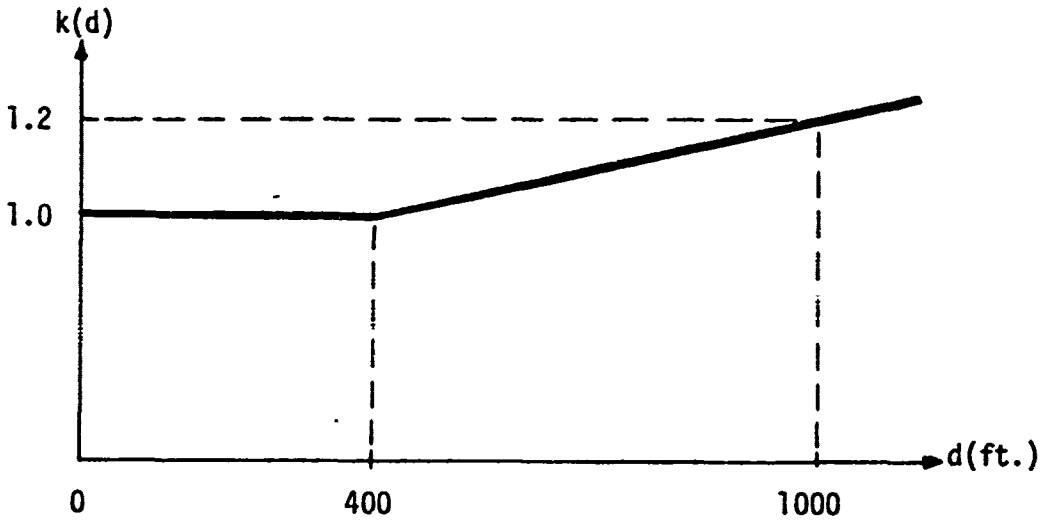


Figure 3.4: Platoon dispersion factor.

Relationships that are different from the one shown in Fig. 3.4 may be taken on specific links, if manifested so by empirical evidence (one different example is demonstrated by the experimental data shown in the next section). q(d) is determined by flow conservation considerations:

$$q(d)\ p(d) = q(0)\ p(0),$$

therefore,

$$q(d) = q(0)\ \frac{p(0)}{p(d)}\ . \qquad (3.20)$$

3.3 Evaluation Results

The traffic flow model described in Section 3.2 was evaluated by comparison with field data on platooned traffic reported by the British Transport and Road Research Laboratory [16]. In this experiment, observations on individual vehicle arrivals were recorded at four different positions downstream of a signal-controlled intersection operating on a 90-second cycle. This information was subsequently processed to determine the average number of arrivals in each two-second time interval, referred with respect to the onset of the green phase at the upstream traffic signal, for each one of the four locations.

The results together with a scaled schematic of the intersection and roadway where the information was collected are shown in Fig. 3.5. The spreading, or dispersion, that takes place throughout the approximate 1000 ft. of roadway downstream of the intersection can be readily seen; e.g., the fourth platoon has 72-sec. passage time that should be compared to the 40-sec. effective green time at the intersection.

The actual platoons were approximated by rectangular-shaped platoons containing the same total number of vehicles. The width was made correspondent to a pair of imaginary rays delineating the trajectories of the leading edge and the trailing edge of the rectangular approximation. The relationships of Section 3.1 were then used to calculate average delay per vehicle in seconds $z(\gamma)$ as a function of arrival time γ, parameterized for three splits g/r for both the actual platoons and their rectangular approximations. The splits were chosen to cover a wide range of possible degrees of saturation. The results are illustrated in Figs. 3.6 to 3.9, corresponding to each of the four different platoons of Fig. 3.5. The following observations can be made: In most cases there exists a close fit between the delay functions derived from the actual platoons and those derived from their rectangular approximations. The largest deviations occur at extremes of splits and/or arrival times (and hence offsets) that are only rarely expected in practice. In some of these cases it is possible to improve the fit through a better selection of the parameters γ and p for the rectangular platoon. Clearly, further experience and field data ought to guide this selection.

3.4 Analysis of the LPF for Rectangular Platoons

In this section further relations are developed to characterize the link performance function $z(\gamma,r)$, based on the particular traffic flow model described above. These relations will be useful in developing the analytics to be employed later in the network optimization program. As was mentioned previously, the approach outlined in this section can be similarly applied to additional measures-of-effectiveness, such as number of stops, acceleration noise, etc.

For a fixed platoon length p, the average delay z is a function of arrival time γ (translatable to offset time ϕ via equation (2.2)) and the split (say r). We can prove the following theorems:

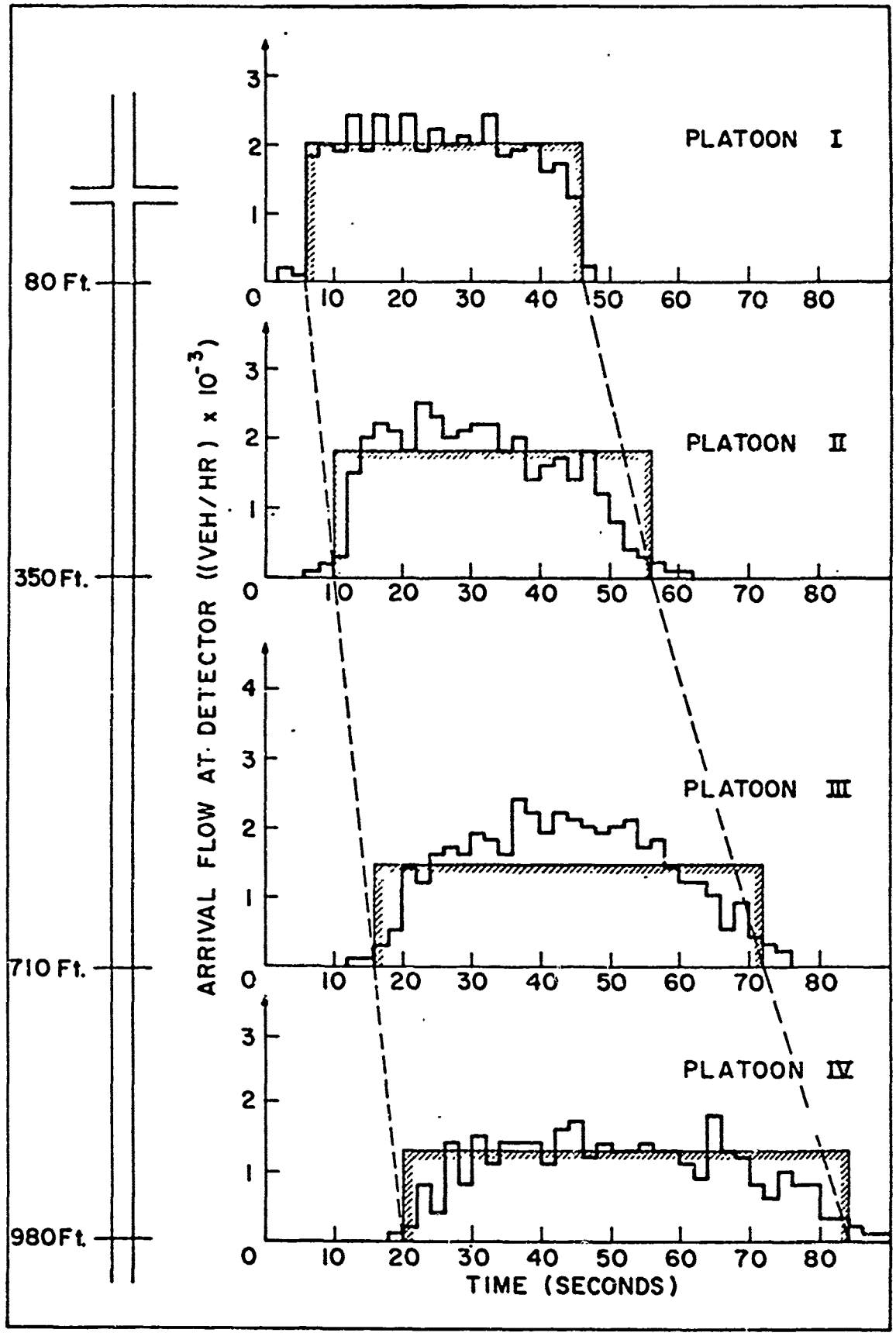
Theorem 3.1: Delay is minimal when the arrival time γ is such that the trailing edge of the platoon arrives at the light just before it turns red and is able to clear the intersection entirely, i.e.,

$$\gamma + p = g. \tag{3.21}$$

Proof: Referring to Fig. 3.10 we distinguish two cases:

(a) $p \leq g$, (b) $p \geq g$.

Figure 3.5: Average platoon flow and rectangular approximation at four observation posts downstream of a signal-controlled intersection.


80 Ft.
350 Ft.
710 Ft.
980 Ft.
PLATOON I
PLATOON II
PLATOON III
PLATOON IV
ARRIVAL FLOW AT DETECTOR ((VEH/HR) x 10^{-3})
TIME (SECONDS)

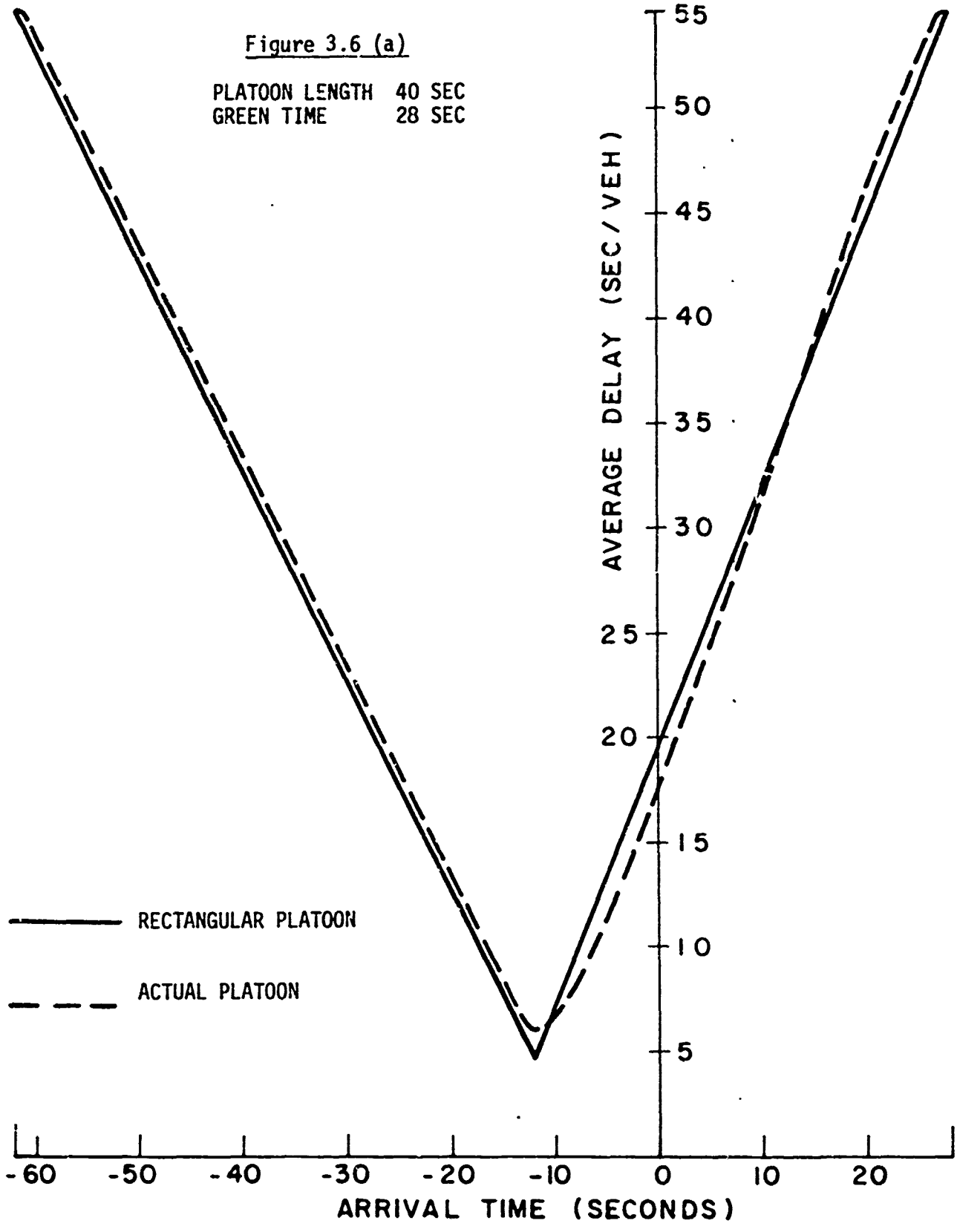
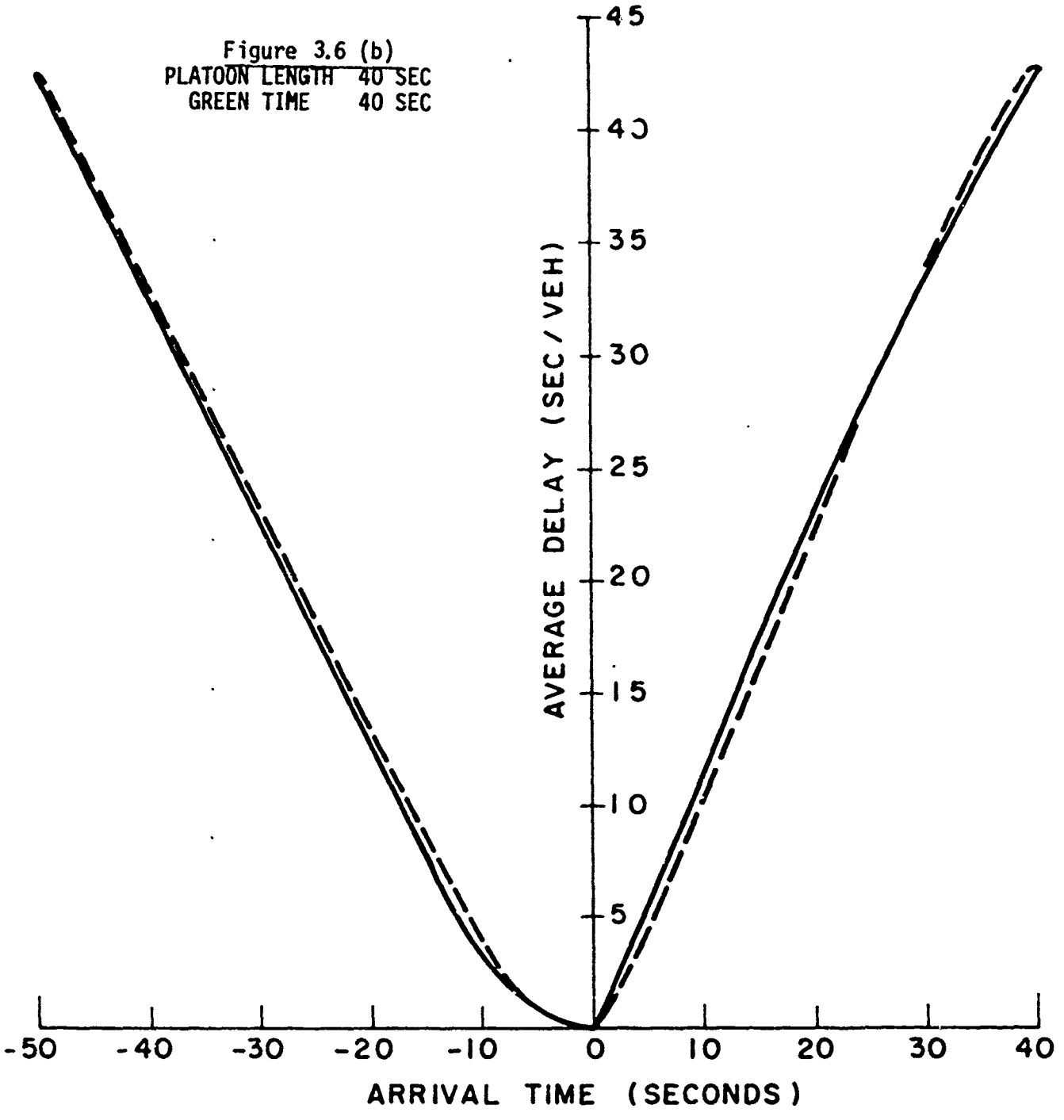


Figure 3.6 (a)
PLATOON LENGTH 40 SEC
GREEN TIME 28 SEC
AVERAGE DELAY (SEC/VEH)
55
50
45
40
35
30
25
20
15
10
5
RECTANGULAR PLATOON
ACTUAL PLATOON
-60
-50
-40
-30
-20
-10
0
10
20
ARRIVAL TIME (SECONDS)

Figure 3.6 (b)
PLATOON LENGTH 40 SEC
GREEN TIME 40 SEC



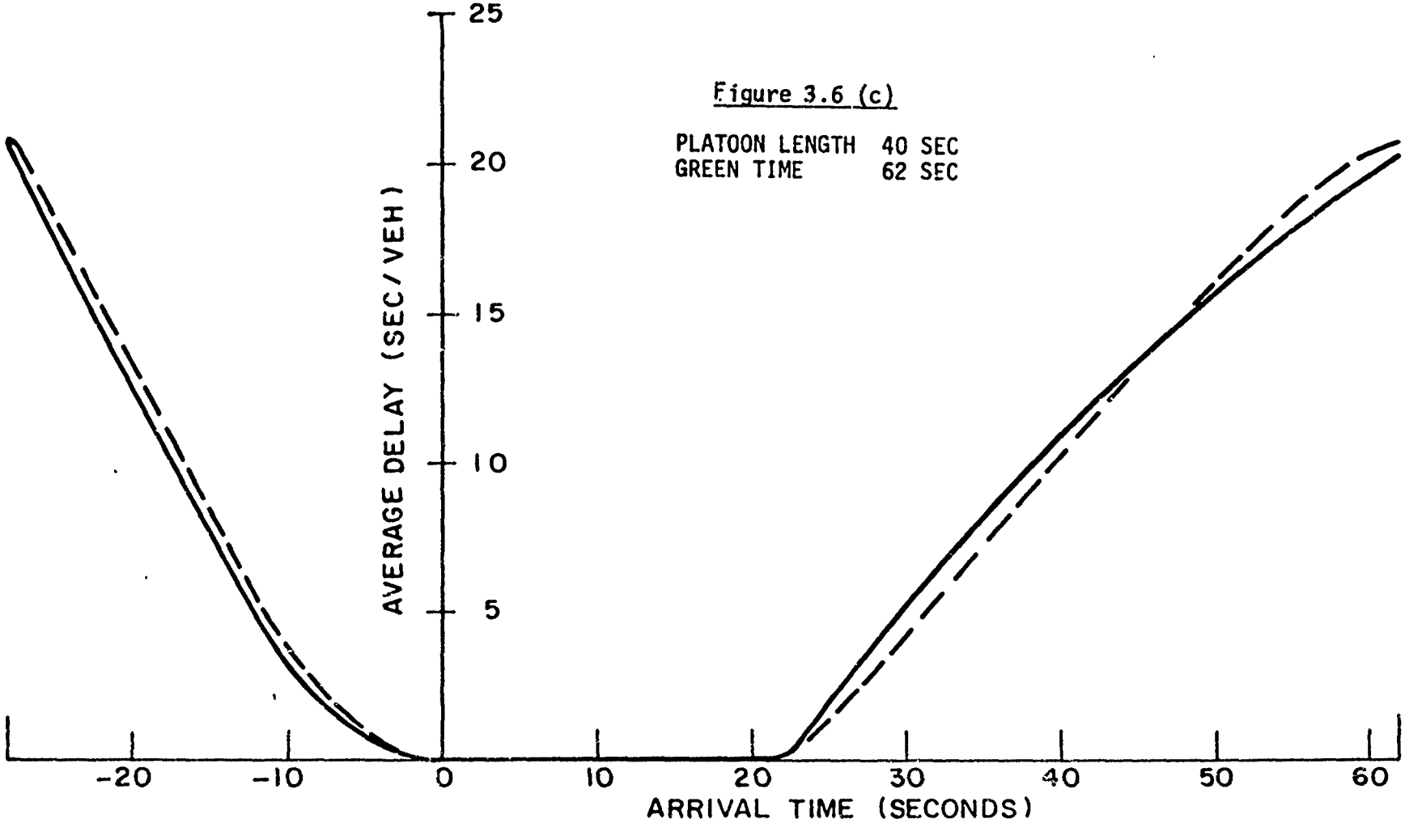
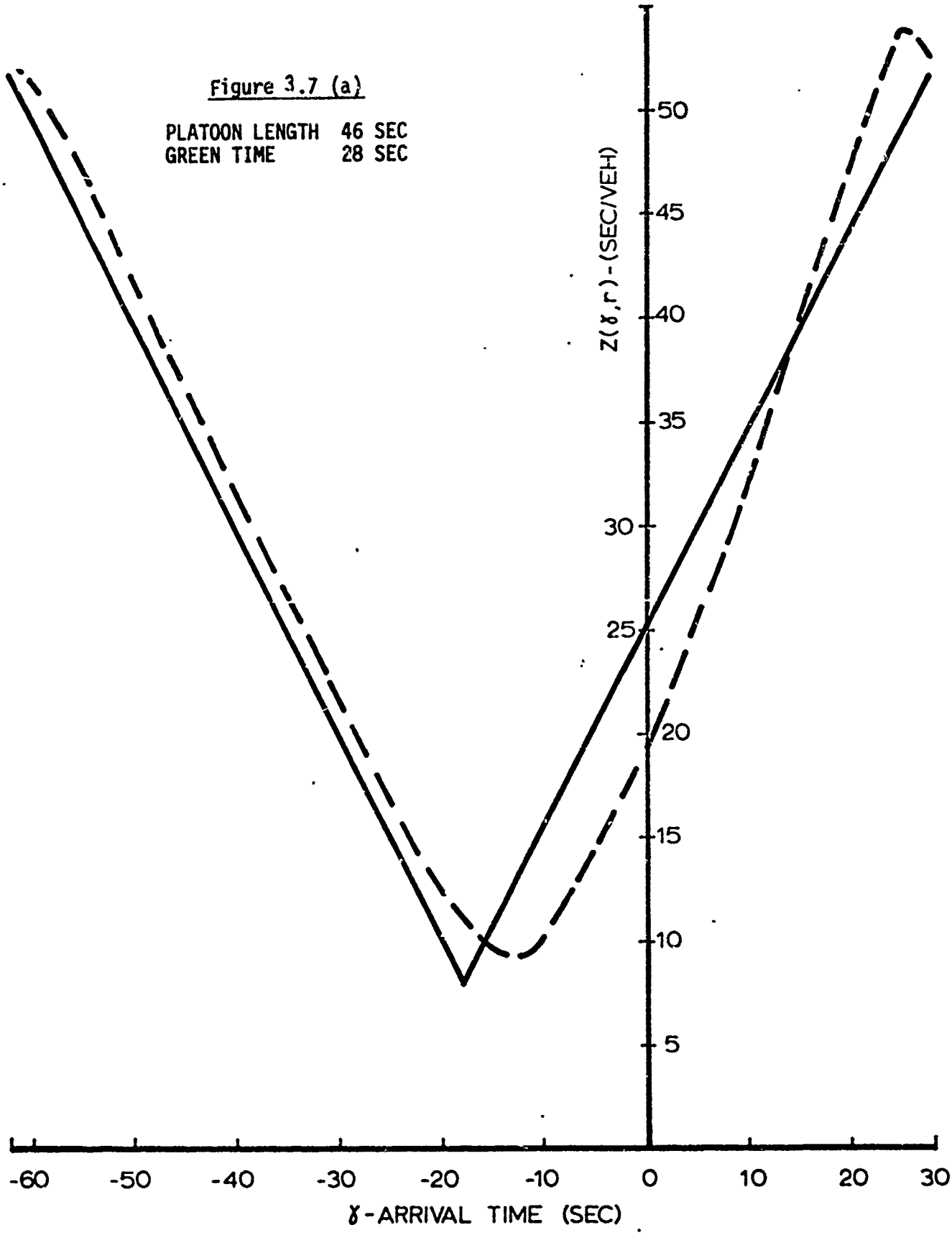


Figure 3.6 (c)

PLATOON LENGTH 40 SEC
GREEN TIME 62 SEC

Figure 3.7 (a)

PLATOON LENGTH 46 SEC
GREEN TIME 28 SEC



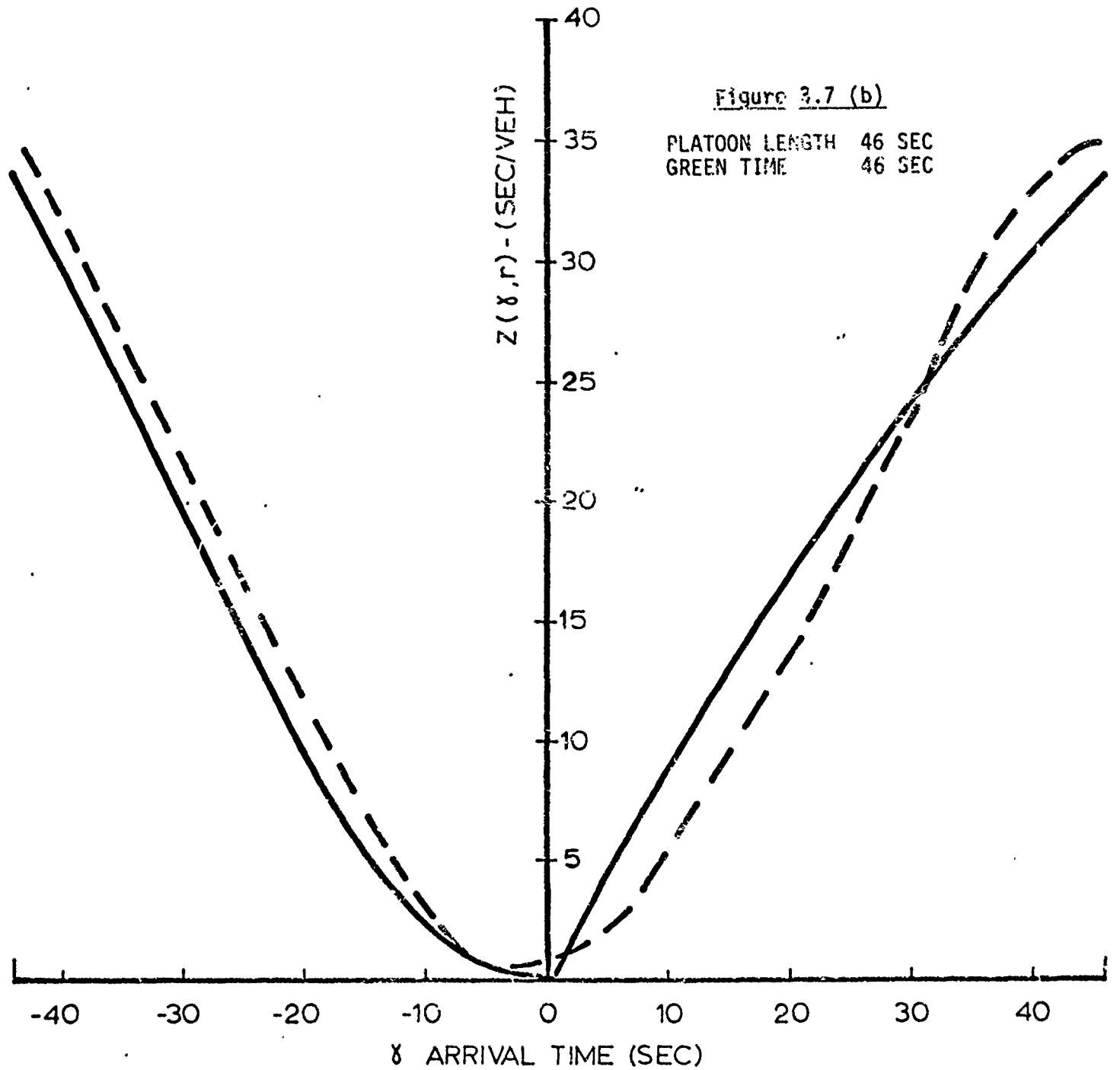
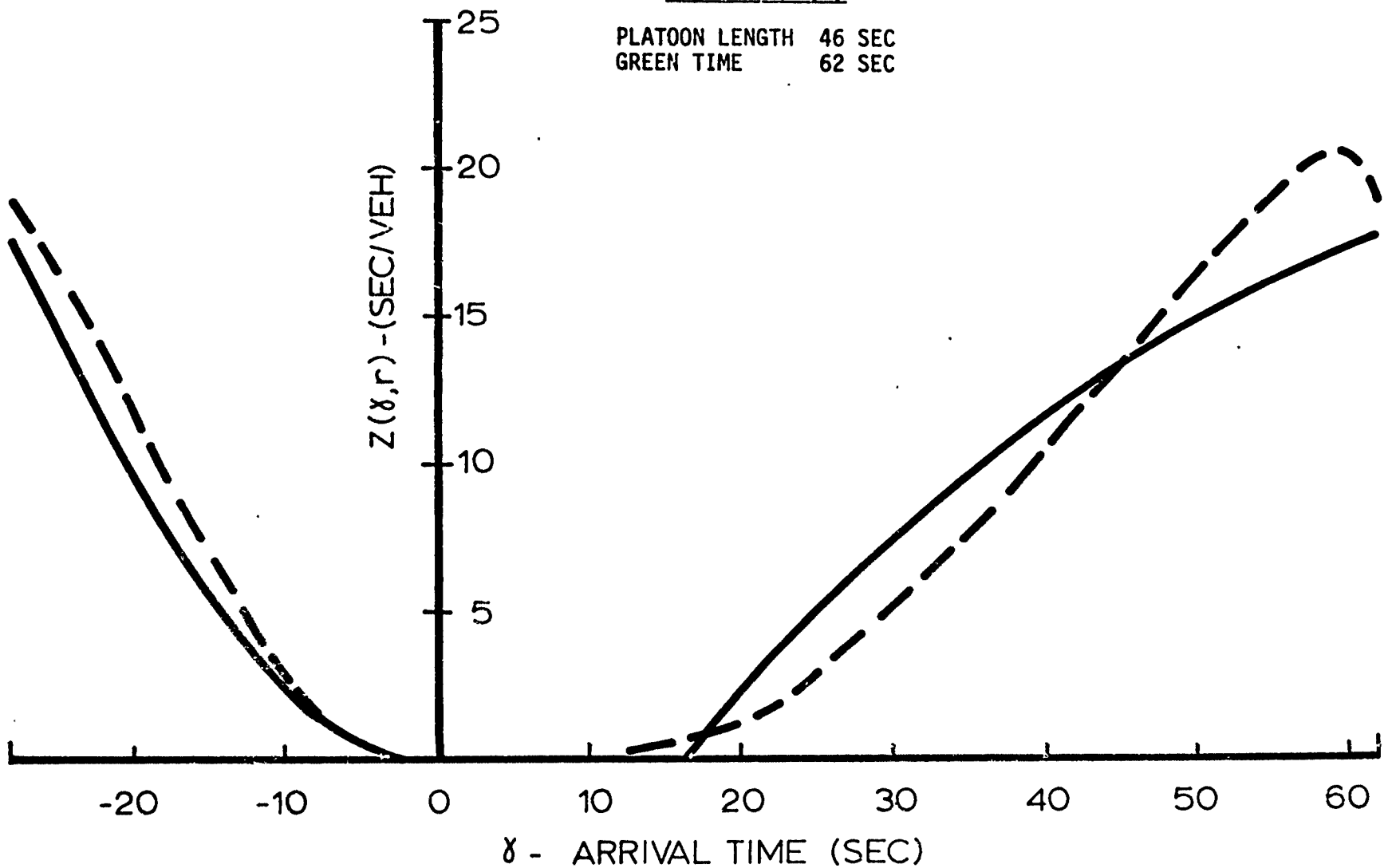


Figure 3.7 (b)
PLATOON LENGTH 46 SEC
GREEN TIME 46 SEC
Z(γ,r) - (SEC/VEH)
40
35
30
25
20
15
10
5
-40
-30
-20
-10
0
10
20
30
40
γ ARRIVAL TIME (SEC)

Figure 3.7 (c)


PLATOON LENGTH 46 SEC
GREEN TIME 62 SEC
Z(γ,r) - (SEC/VEH)
25
20
15
10
5
-20
-10
0
10
20
30
40
50
60
γ - ARRIVAL TIME (SEC)

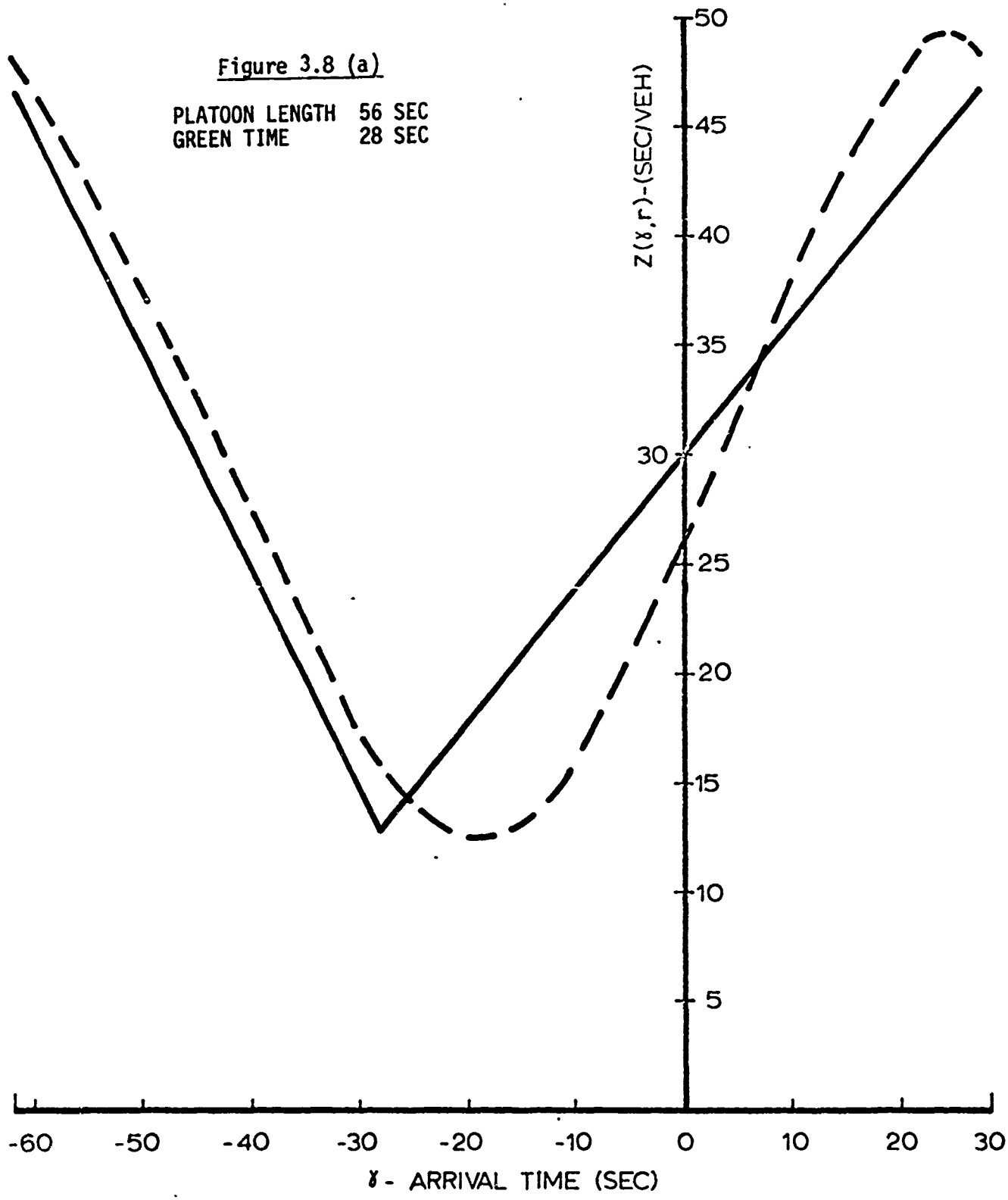


Figure 3.8 (a)
PLATOON LENGTH 56 SEC
GREEN TIME 28 SEC
Z(γ,r)-(SEC/VEH)
50
45
40
35
30
25
20
15
10
5
-60
-50
-40
-30
-20
-10
0
10
20
30
γ - ARRIVAL TIME (SEC)

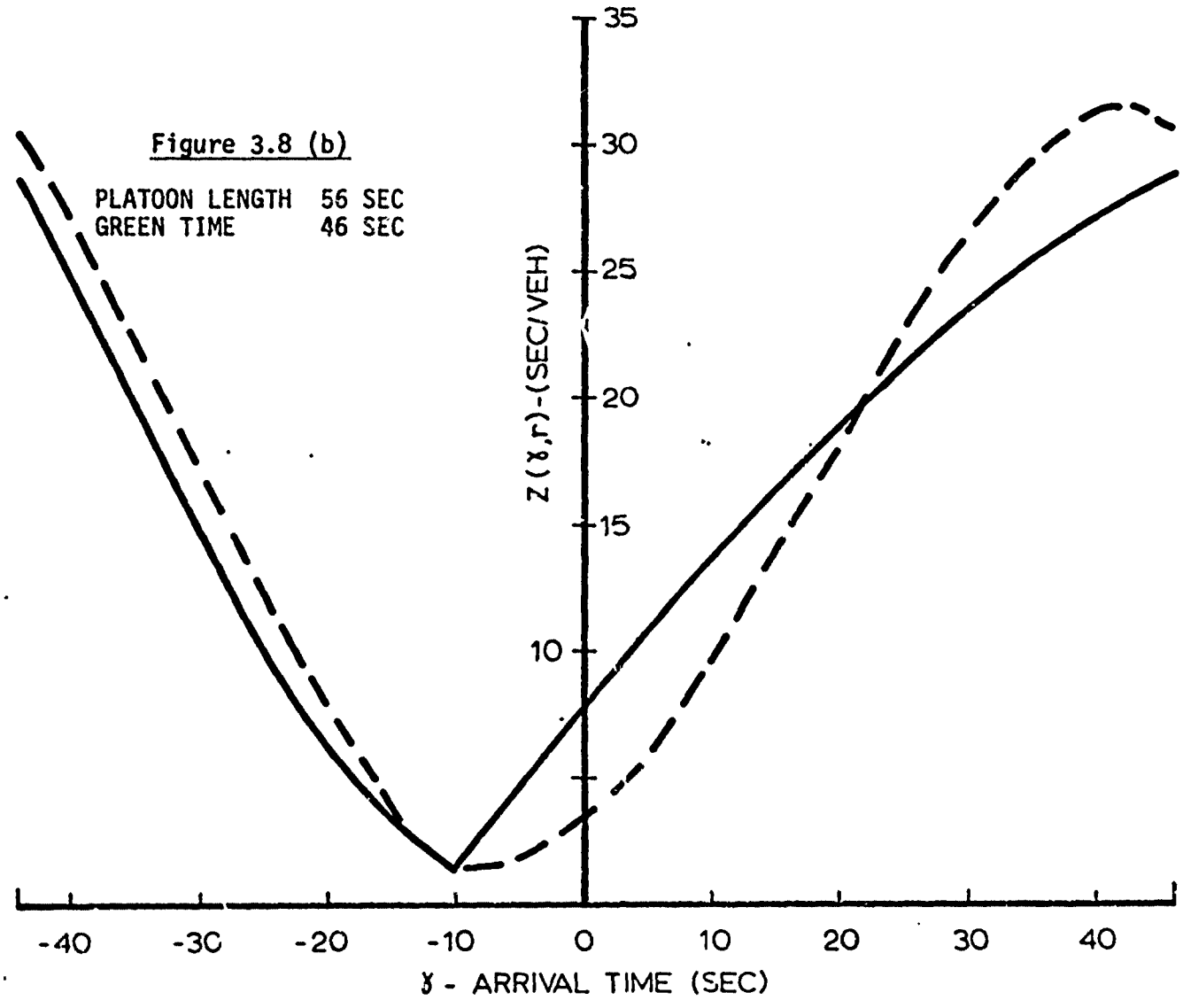


Figure 3.8 (b)
PLATOON LENGTH 56 SEC
GREEN TIME 46 SEC
35
30
25
20
15
10
Z(γ,r)-(SEC/VEH)
-40
-30
-20
-10
0
10
20
30
40
γ - ARRIVAL TIME (SEC)

Figure 3.8 (c)

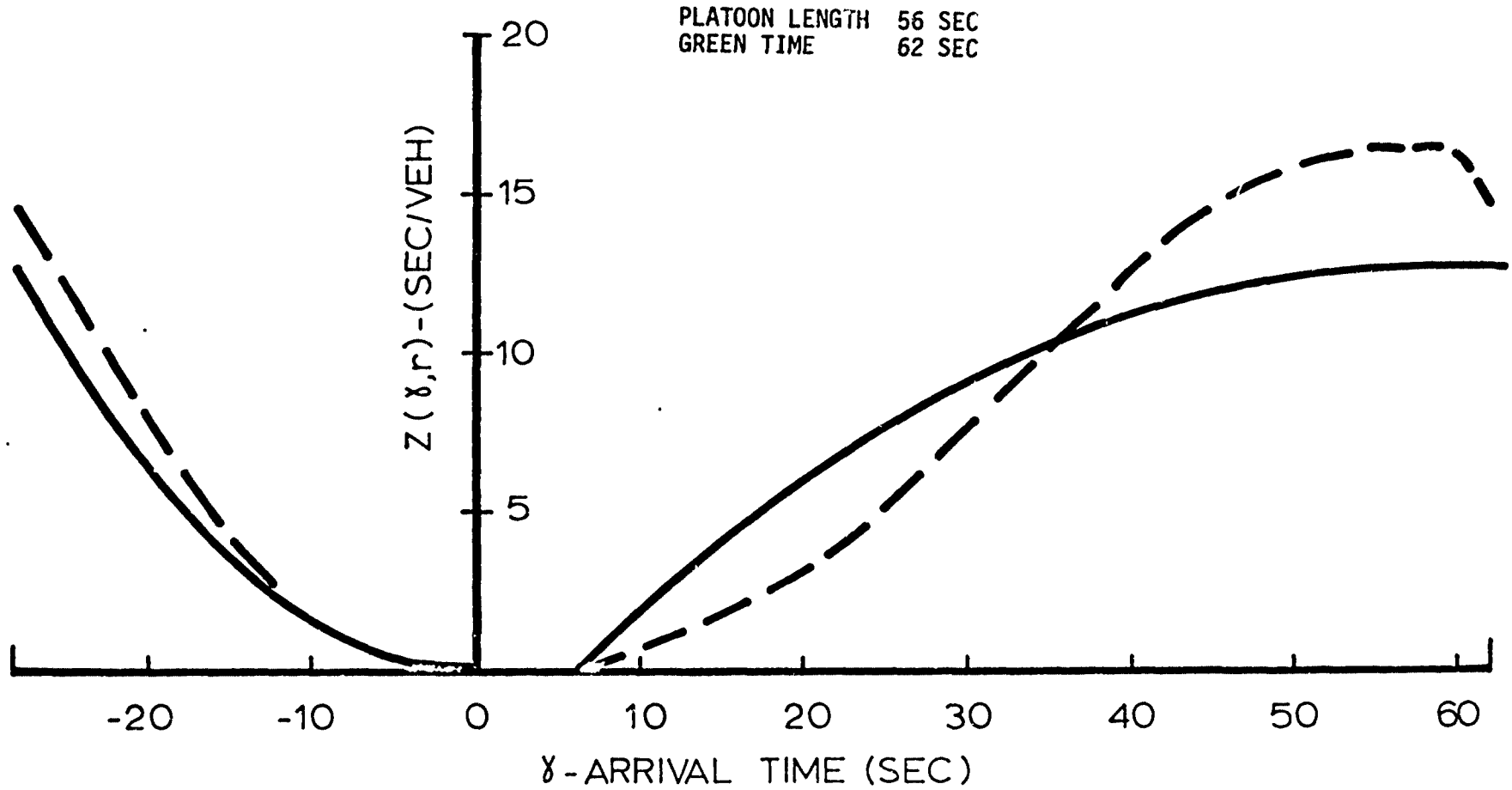
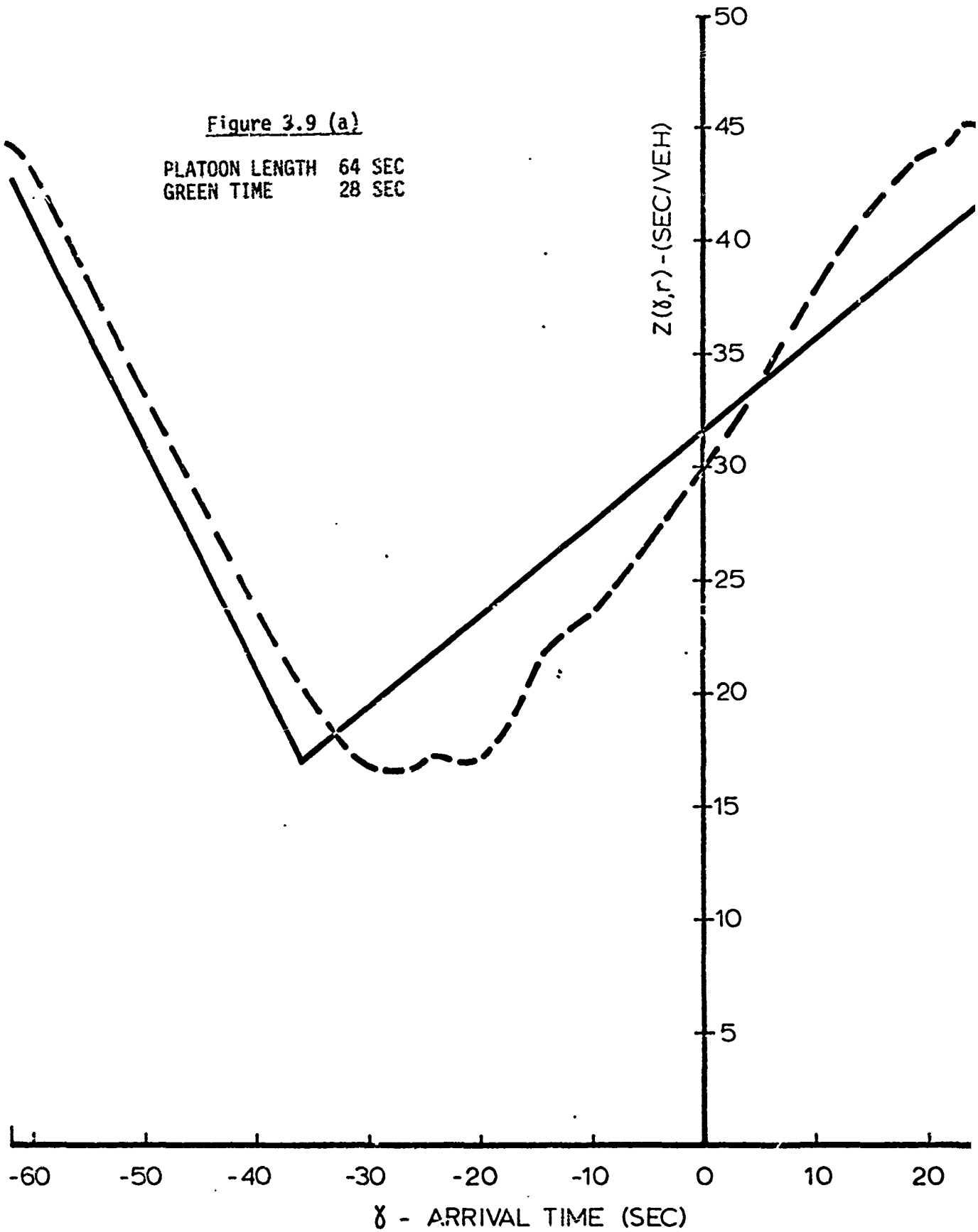

PLATOON LENGTH 56 SEC
GREEN TIME 62 SEC
20
15
10
5
Z(Y,r)-(SEC/VEH)
-20
-10
0
10
20
30
40
50
60
Y-ARRIVAL TIME (SEC)

Figure 3.9 (a)

PLATOON LENGTH 64 SEC
GREEN TIME 28 SEC



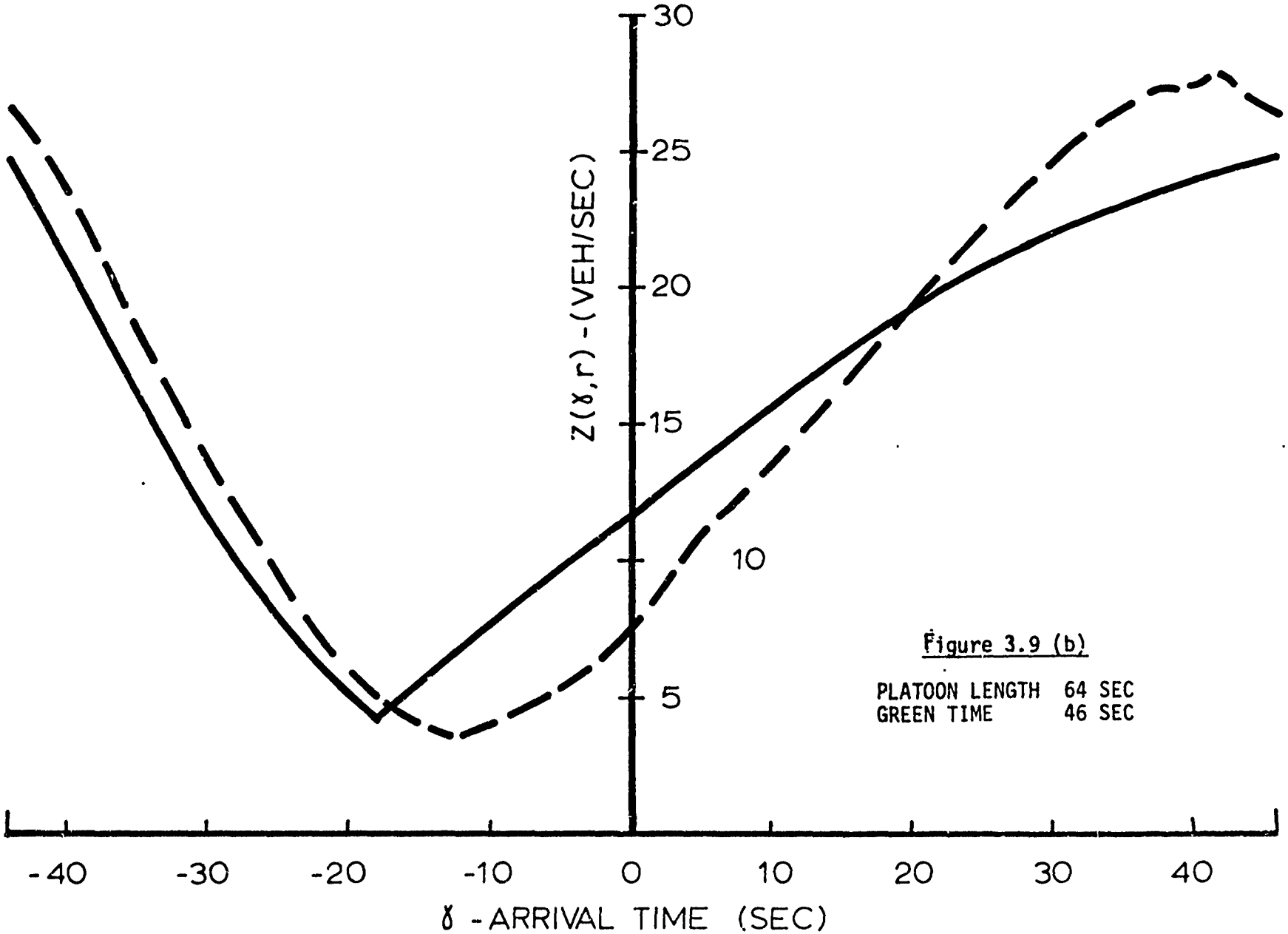
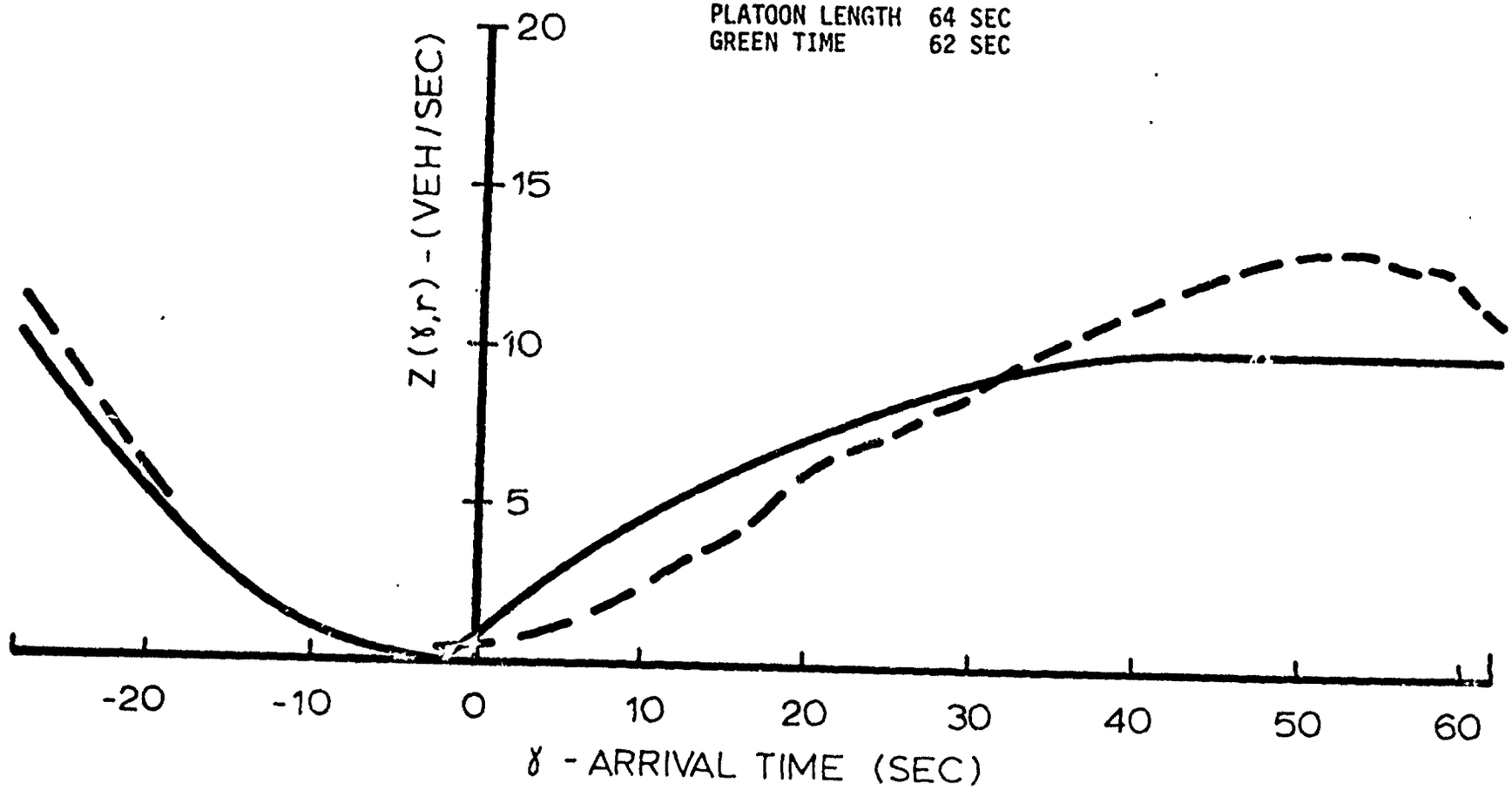


Figure 3.9 (b)

PLATOON LENGTH 64 SEC
GREEN TIME 46 SEC

Figure 3.9 (c)

PLATOON LENGTH 64 SEC
GREEN TIME 62 SEC



Case a: A degeneracy occurs for $\gamma \in [0, g-p]$. The platoon fits into the green opening and since no cars are stopped the delay is zero (case a.1). For $\gamma < 0$ we have a headstopping effect: $q(-\gamma)$ cars arrive during red and the delay incurred is indicated by the dashed area shown in case a.2. For $\gamma + p > g$, a portion p_x of the platoon is tailchopped by the red light: qp_x cars are stopped and delayed (case a.3).

Case b: When $\gamma + p = g$, i.e., $\gamma = -(p-g)$, $q(-\gamma)$ are stopped. The delay is indicated by the dashed area shown in case b.1. For $\gamma < -(p-g)$ headstopping increases and delay increases correspondingly (case b.2). For $\gamma > -(p-g)$ the total number of cars that arrive during red is the same as in case b.1, i.e., $q(p-g)$. However, in this case both headstopping and tailchopping occur: qp_x are stopped right after the light turns red and thus the total delay is larger than in b.1 (case b.3). Q.E.D.

Referring to Fig. 3.10 (case b.2) we calculate the total delay $Z(\gamma)$ incurred by the platoon. It is the sum of two areas:

$$Z(\gamma) = A_1 + A_2 = \frac{1}{2} q\gamma^2 + \int_0^{t_o} [q(t-\gamma) - st]dt.$$

The queue clearance time t_o is determined by the identity

$$Q(t_o) = q(-\gamma) + qt_o = st_o = 0,$$

following eqn. (3.8). Therefore,

$$t_o = \frac{q\gamma}{q-s} = -\gamma \frac{y}{1-y}, \tag{3.22}$$

where $y = q/s$ denotes the ratio of platoon flow to saturation flow on a particular link. Hence,

$$A_2 = [(q - s) \frac{t^2}{2} - q\gamma t]_o^{t_o} = -\frac{1}{2} \frac{q^2\gamma^2}{q-s}$$

and,

$$Z(\gamma) = \frac{1}{2} q\gamma^2 - \frac{1}{2} \frac{q^2\gamma^2}{q-s} = \frac{1}{2} \frac{qs\gamma^2}{s-q}$$

The average delay is given by

$$z(\gamma) = \frac{Z(\gamma)}{qp} = \frac{\gamma^2}{2p} \cdot \frac{1}{1-y}. \tag{3.23}$$

a. $p < g$

a.1 $\gamma > 0$
$\gamma + p = g$

a.2 $\gamma < 0$
$\gamma + p > 0$

a.3 $\gamma > 0$
$\gamma + p > g$
$p_x = \gamma + p - g$

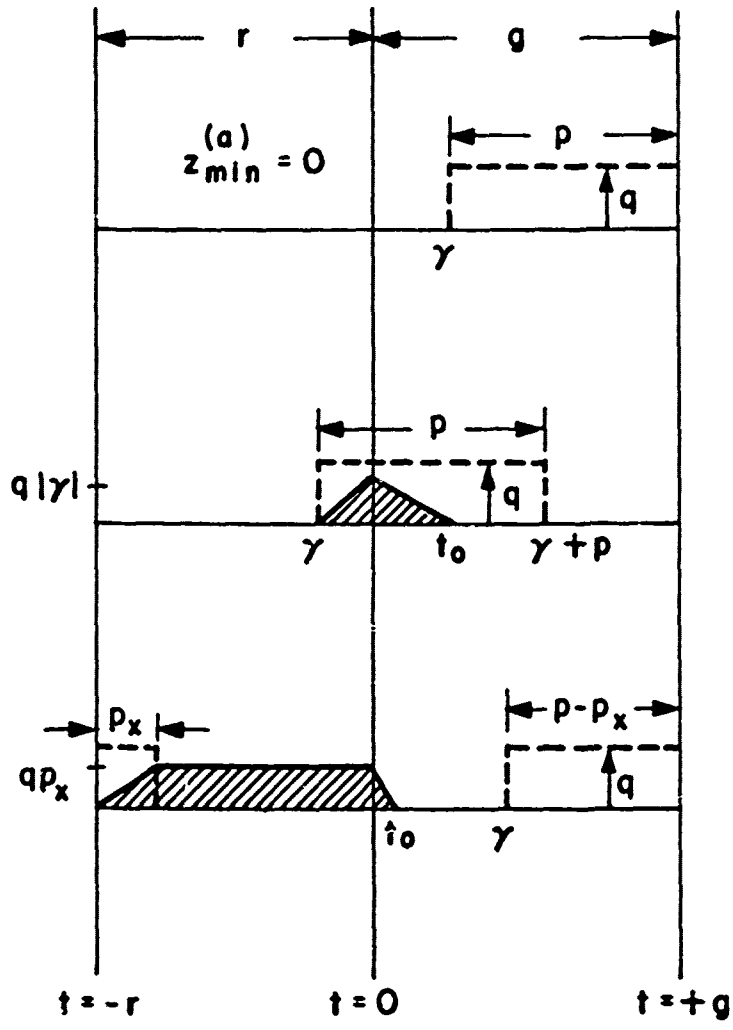


Figure 3.10-(a).

b. $p > g$

1. $\gamma < 0$
   $\gamma + p = g$

2. $\gamma < 0$
   $\gamma + p < g$

3. $\gamma + p < g$
   $p_x = \gamma + p - g$

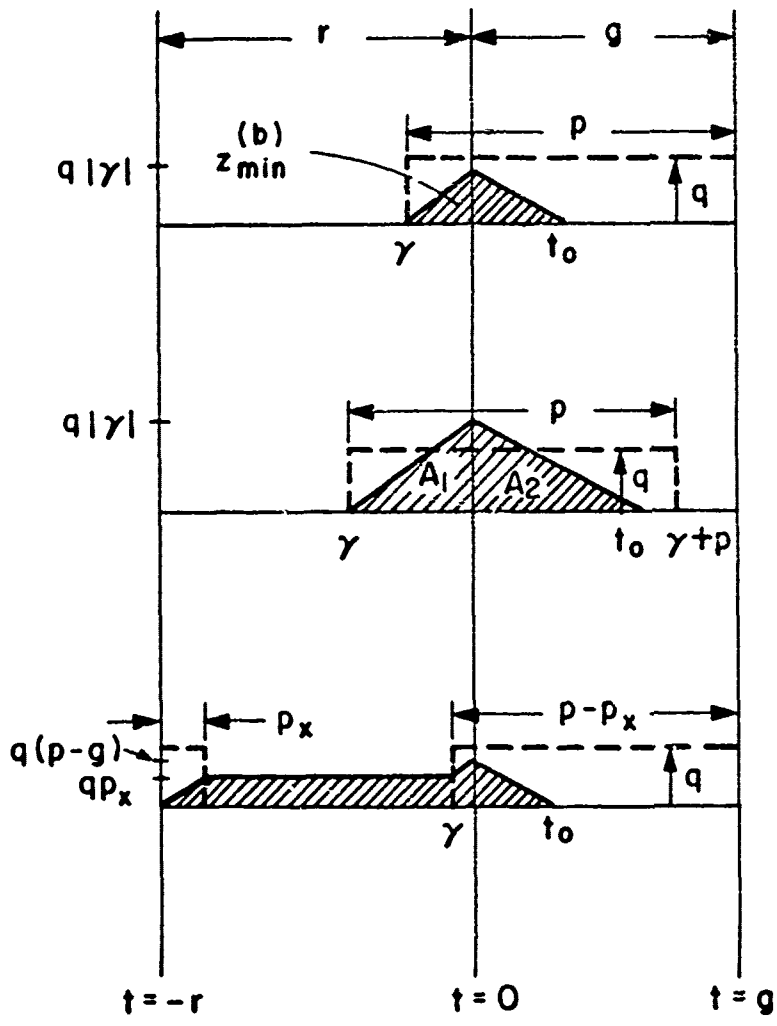


Figure 3.10(b)

By Theorem 3.1 the minimal value occurs when $\gamma = \underline{\gamma} = -(p-g)$. Therefore,

$$z_{min} = \frac{\underline{\gamma}^2}{2p} \cdot \frac{1}{1-y} = \frac{(p-g)^2}{2p} \cdot \frac{1}{1-y} \tag{3.24}$$

Theorem 3.2: Delay is maximal when the arrival time γ is such that the leading edge of the platoon arrives at the signal stop line right after the light turns red, i.e.,

$$\gamma = \bar{\gamma} = -r \tag{3.25}$$

Proof: Referring to Fig. 3.11 we distinguish three cases:

(a) $p \leq r$, (b) $r < p \leq t_o + r$, (c) $p > t_o + r$.

Case a: Fig. a.1 shows the queue evolvement and total delay for $\gamma = -r$. For $\gamma > -r$ (case a.2) the platoon arrives later during the red period and the stopped cars wait a shorter time for the green light than in case a.1. Similarly, for $\gamma > 0$, a portion $(g-\gamma)$ of the platoon clears the intersection before red starts and delay is reduced in this case as well (case a.3).

The total delay in case a can be described as the summation of three sub-areas:

$$A_1 = \frac{1}{2} qp^2; \; A_2 = -pq(\gamma+p); \; A_3 = \int_0^{t_o} (pq - st)dt.$$

The clearance time t_c is determined by the identity $pq = st_o$; i.e.,

$$t_o = \frac{q}{s}p = yp \tag{3.26}$$

Consequently, $A_3 = \frac{1}{2}qyp^2$ and the total delay is,

$$Z(\gamma) = A_1 + A_2 + A_3 = qp(\frac{yp}{2} - \frac{p}{2} - \gamma).$$

Following eqn. (3.10),

$$z^{(a)}(\gamma) = \frac{p}{2}(y-1) - \gamma \tag{3.27}$$

indicating that the average delay decreases linearly as γ is increased. The maximum delay occurs at $\bar{\gamma} = -r$,

a. $p \leq r$

a.1 $\gamma = -r$

a.2 $\gamma > -r$

a.3 $\gamma > 0$
$p_x = \gamma + p - g$

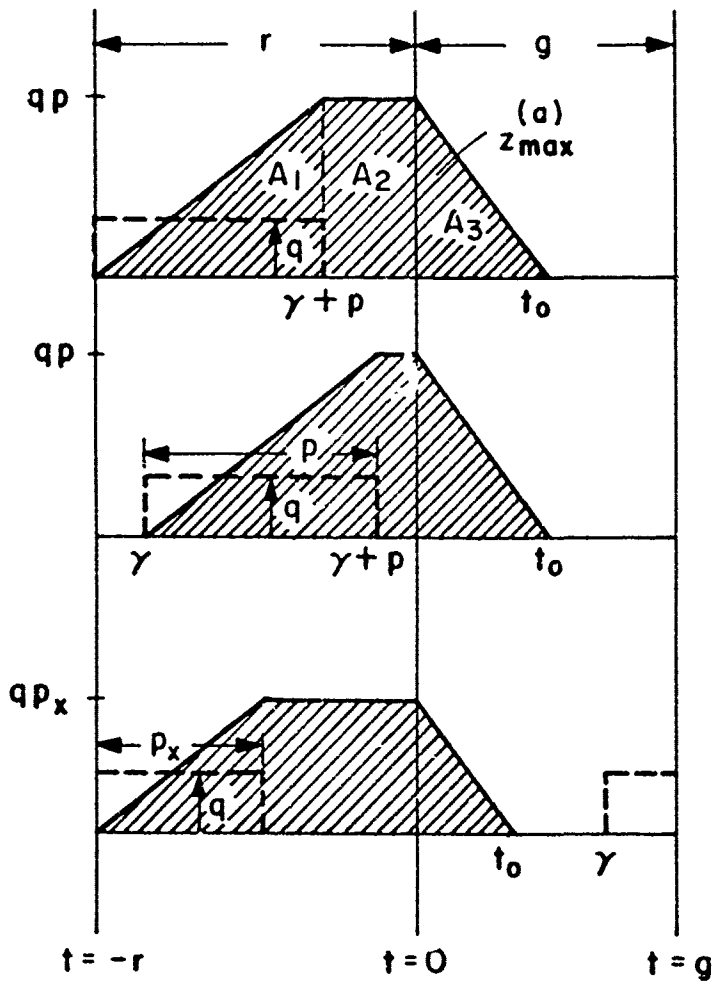


Figure 3.11-(a)

b. $r < p \leq t_o + r$

b.1 $\gamma = -r$

b.2 $\gamma > -r$

b.3 $\gamma > 0$
$p_x = \gamma + p - g$

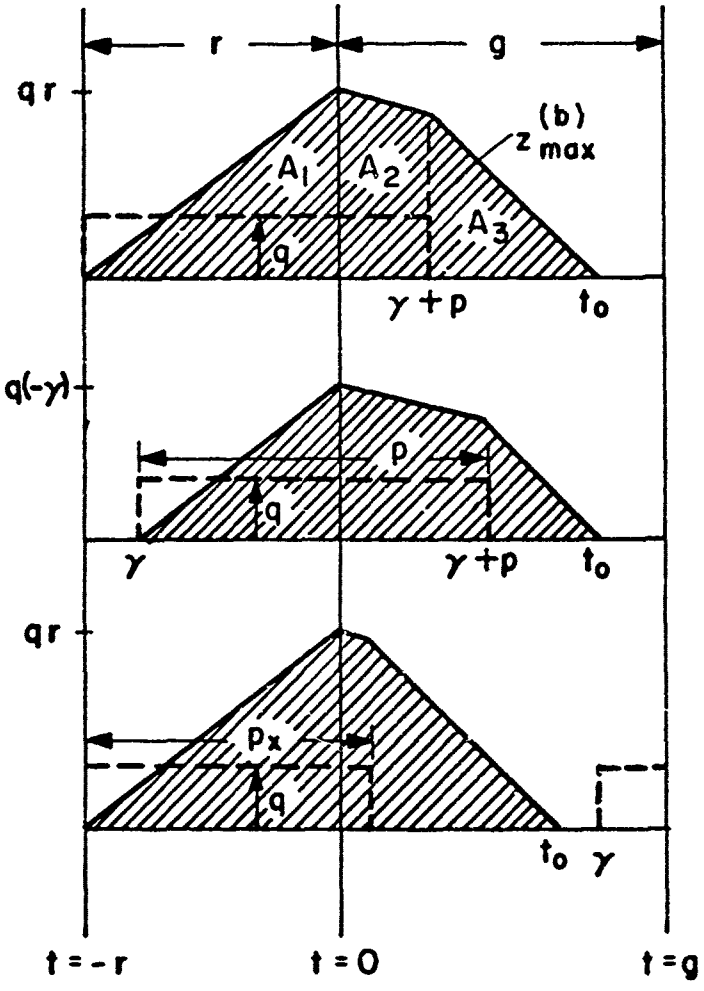


Figure 3.11-(b)

c. $p > r / (1 - y)$

c.1 $\gamma = -r$

c.2 $\gamma > -r$

c.3 $p_x = \gamma + p - g$
$p_x > r/(1-y_o)$

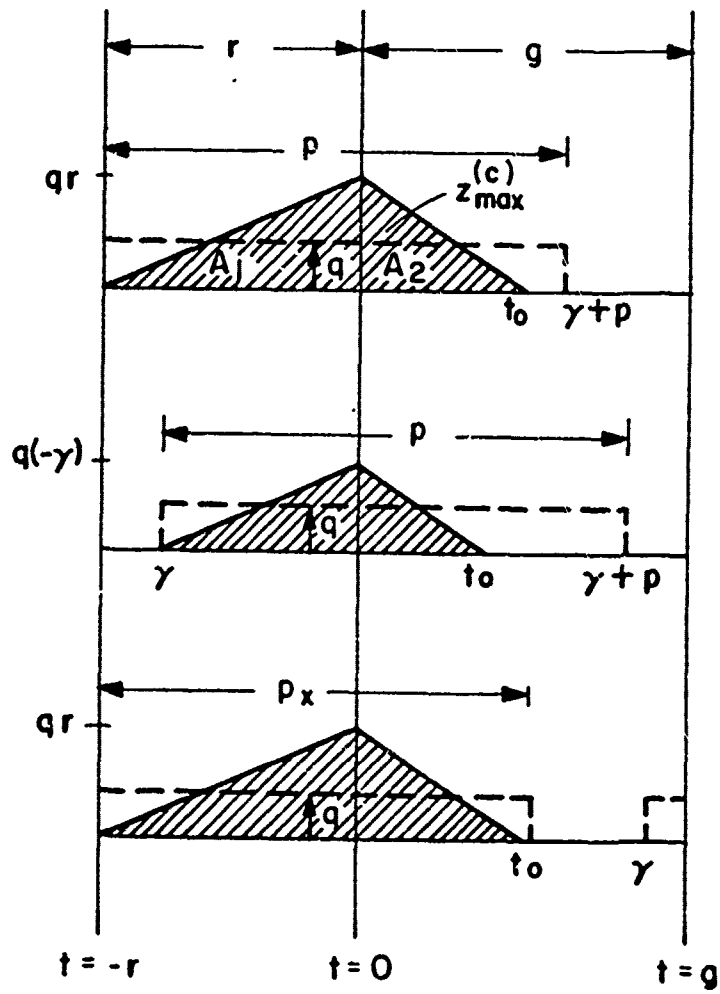


Figure 3.11-(c)

$$z_{max}^{(a)} = \frac{p}{2}(y-1) + r \tag{3.28}$$

Case b: Analysis is similar to that of case a(see Fig. 3.11 - (b)).

The maximal total delay in this case is again a summation of three sub-areas, where

$$A_1 = \int_{\gamma}^{0} q(t-\gamma)dt = \frac{1}{2}q\gamma^2$$

$$A_2 = \int_{0}^{p+\gamma} [q\ (t-\gamma) - st]dt = (q-s)\ \frac{(p+\gamma)^2}{2} - q\gamma(p+\gamma)$$

$$A_3 = \int_{p+\gamma}^{t_o} [Q_{t=p+\gamma} - s(t-p-\gamma)]dt.$$

The queue length at $t = p+\gamma$ is $Q_{t=p+\gamma} = qp - s(p+\gamma)$ and $t_o = yp$ as in case a. Consequently,

$$A_3 = \frac{1}{2}\ \frac{(qp)^2}{s} - qp(p+\gamma) + \frac{1}{2}s(p+\gamma)^2.$$

By addition we obtain,

$$Z(\gamma) = \frac{1}{2}\ \frac{(qp)^2}{s} - \frac{1}{2}q(p+\gamma)^2 + \frac{1}{2}q\gamma^2 \tag{3.29}$$

and,

$$z^{(b)}(\gamma) = \frac{1}{2p}[yp^2 - (p+\gamma)^2 + \gamma^2] = \frac{p}{2}(y-1) - \gamma = z^{(a)}(\gamma). \tag{3.30}$$

Therefore, for $\gamma = \bar{\gamma} = -r$ we have

$$z_{max}^{(a)} = z_{max}^{(b)} = z_{max} \tag{3.31}$$

Case c: $p > t_o + r$, i.e., flow is such that arrivals extend beyond the queue clearance time. Substituting $t_o = yp$ we obtain

$$p > \frac{r}{1-y} \tag{3.32}$$

(The same condition is obtained when substituting expression (3.33) derived for t_o below.)

Analysis is again similar to cases (a) and (b). Referring to Fig. 3.11 - (c), total delay is a sum of two areas:

$$A_1 = \int_{\gamma}^{0} q(t-\gamma)dt = \frac{1}{2}q\gamma^2$$

$$A_2 = \int_{0}^{t_o} [q(t-\gamma) - st]dt = \frac{1}{2}q\gamma^2 \cdot \frac{y}{1-y}.$$

t_o is determined according to eqn. (3.8):

$$q\gamma + qt_o - t_o s = 0$$

$$t_o = \frac{q}{s-q}\gamma = \frac{y}{1-y}\gamma . \tag{3.33}$$

By addition we obtain,

$$Z(\gamma) = A_1 + A_2 = \frac{1}{2}q\gamma^2 \cdot \frac{1}{1-y} \tag{3.34}$$

and,

$$z(\gamma) = \frac{\gamma^2}{2p} \cdot \frac{1}{1-y}. \tag{3.35}$$

To obtain the maximal average delay for this case, we substitute $\bar{\gamma} = -r$:

$$z_{max}^{(c)} = \frac{r^2}{2p} \cdot \frac{1}{1-y}. \tag{3.36}$$

Q.E.D.

Referring to Fig. 3.11-(c.3) we observe that a degenerate case occurs for $p_x > r/(1-y)$, where $p_x = \gamma+p-g$. $z_{max}^{(c)}$ maintains a constant value in this case for as long as

$$C-p + r\frac{y}{1-y} \leq \gamma \leq C-r. \tag{3.37}$$

The different cases illustrated in Fig. 3.10 and Fig. 3.11 do not cover all the possible combinations of arrival time, platoon length, split value, flow and saturation flow. Two generalized analytical expressions for delays incurred by platoons at traffic signals are given in Ref. [2].

We are now able to deduce the following two corollaries, confining ourselves to the span of one cycle time, i.e., $\gamma \in [-r,+g]$ (see Fig. 3.12):

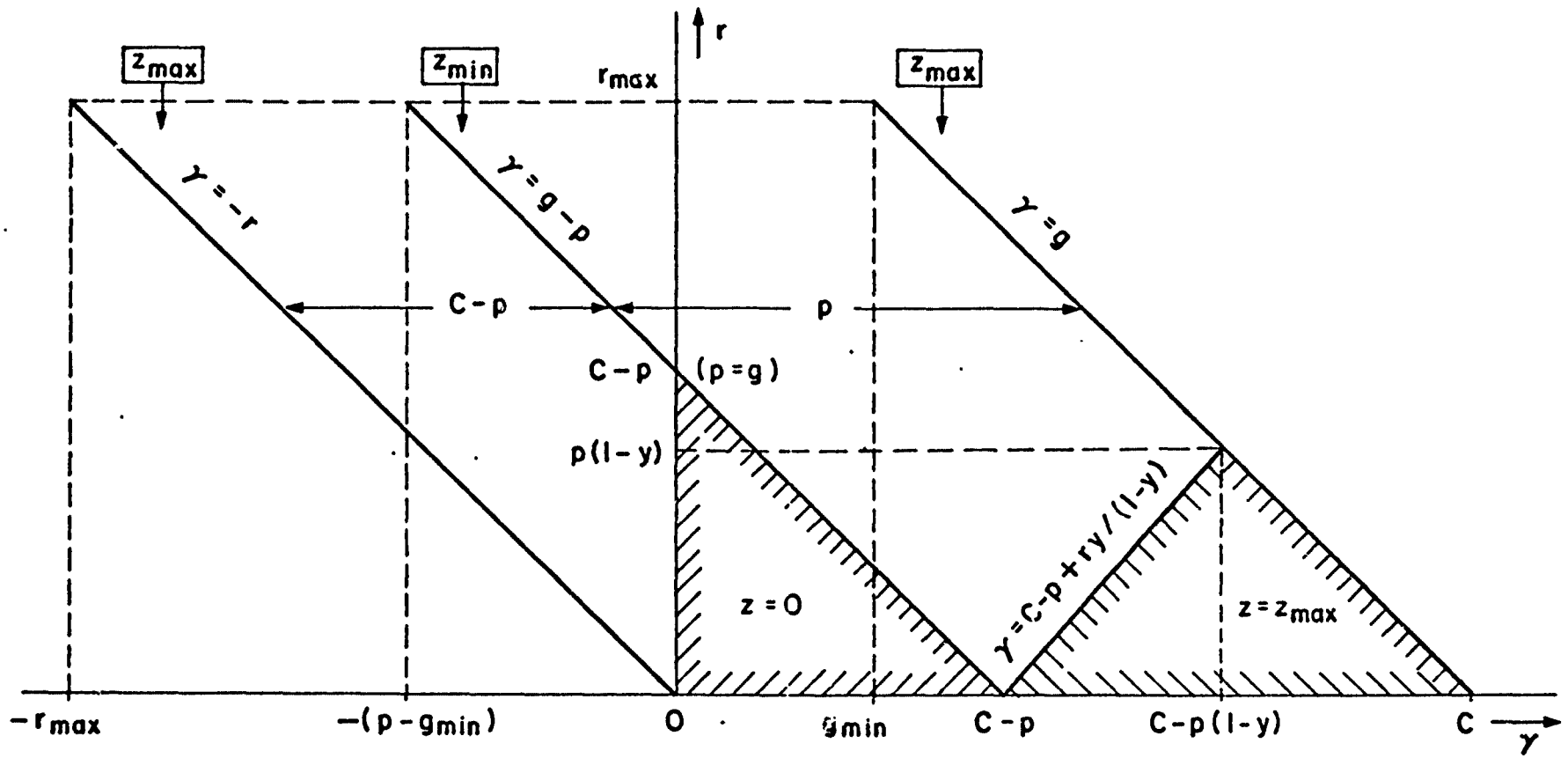


Figure 3.12: Loci for the minimal and maximal platoon delay in (γ,r) plane.

Corollary 1: The locus of minimal delay in plane (γ,r) is the line $\gamma = g-p = C-r-p$. Also, delay is minimal (and equal to zero) in the area circumscribed by $0 \leq \gamma < C-r-p$.

Corollary 2: The locus of maximal delay in plane (γ,r) are the two lines $\gamma = -r$ and $\gamma = g = C-r$, since,

$$z_{max} = \lim_{\gamma \to -r+} z(\gamma,r) = \lim_{\gamma \to +g} (\gamma,r). \qquad (3.38)$$

As shown above, a degeneracy occurs when $p > r(1-y)$. For any $\gamma \in [g - (p-(r+t_0)),g]$ there is no change in delay (at a fixed split) and its value is maximal according to equation (3.31).

Theorem 3.3: For $\gamma \in [-r,0]$ the delay function $z(\gamma,r)$ is a strictly decreasing function of γ.

Proof: Referring to Fig. 3.11 (cases a.2,b.2 and c.2), any positive shift in γ from γ_1 to γ_2, such that $\gamma_2 > \gamma_1$, results in a queue-length curve $Q_2(t)$ that is completely encompassed by $Q_1(t)$. Therefore, $z(\gamma_2,r) < z(\gamma_1,r)$.

Theorem 3.4: For $\gamma \in [0,g]$ the delay function $z(\gamma,r)$ is a non-decreasing function of γ.

Proof: We distinguish the following cases:

p	$<r/(1-y)$	$>r/(1-y)$
$>g$	I	III
$<g$	II	IV

I. Referring to Fig. 3.10 (cases a.3 and b.3), any positive shift in γ from γ_1 to γ_2, such that $\gamma_2 > \gamma_1$, results in a queue-length curve $Q_2(t)$ that completely encompasses $Q_1(t)$. Therefore, $z(\gamma_2,r) > z(\gamma_1,r)$ and $z(\gamma,r)$ is strictly increasing.

II. $z(\gamma,r)$ has a zero-valued flat portion for $\gamma \in [0,g-p]$ and is strictly increasing for $\gamma \in [g-p,g]$.

III. $z(\gamma,r)$ is strictly increasing for $\gamma \in [0,C-p+ry/(1-y)]$ and has a maximal flat portion (variable with split but fixed at any fixed split) for $\gamma \in [C-p+ry/(1-y),g]$.

IV. $z(\gamma,r)$ is strictly increasing for $\gamma \in [g-p,C-p+ry/(1-y)]$ and has two flat portions as in (II) and (III) above.

Theorem 3.5: If $p > r/(1-y)$ then for $\gamma \in [\underline{\gamma},\bar{\gamma}]$ delay is independent of split.

Proof: The theorem is proved most easily with reference to Fig. 3.13. The total delay incurred by the platoon in cases (a) and (b) depends only on the characteristics of the platoon itself (p and g) and its arrival time γ. Since all these parameters are identical in both cases (a) and (b), the resultant queue-length curve, and hence the total delay, are also identical, regardless of the split value (provided, of course, we do not exceed capacity).

Special Case: $y > 1$

The discussion in this section assumed up to now that $y < 1$, i.e., $q < s$. Though rare, a situation may occur in which $y > 1$ on a particular link. For instance, in case a two-lane link is feeding traffic onto a single-lane link (the bottleneck occurs at the intersection connecting them).

The capacity constraint requires as before:

$$qp \leq gs \quad \text{i.e.} \quad g \geq yp.$$

Clearly, since $y > 1$, g must be larger than p.

(Caution: Condition (3.32) on p.41 now has to be read $p(1-y) > r$, since dividing by 1-y which is now negative would change the sense of inequality.)

Maximal delay is given by equations (3.28) or (3.36) as above. However, calculation of minimal delay is different. Referring to Fig. 3.14 we have a queue build-up even though arrivals occur during green only. Total delay is a sum of the two areas A_1 and A_2:

$$A_1 = \int_{\gamma}^{\gamma+p} (q-s)(t-\gamma)dt = (q-s)\frac{p^2}{2}$$

$$A_2 = \int_{\gamma+p}^{t_o} [(q-s)p - s(t-\gamma-p)]dt = (q-s)\frac{p^2}{2}(y-1)$$

where t_o is determined as follows:

$$\begin{aligned} q \cdot p - s(t_o\gamma) &= 0 \\ t_o &= \gamma+yp \end{aligned} \tag{3.39}$$

Consequently,

$$Z_{min} = \tfrac{1}{2}p^2y(q-s) \tag{3.40}$$

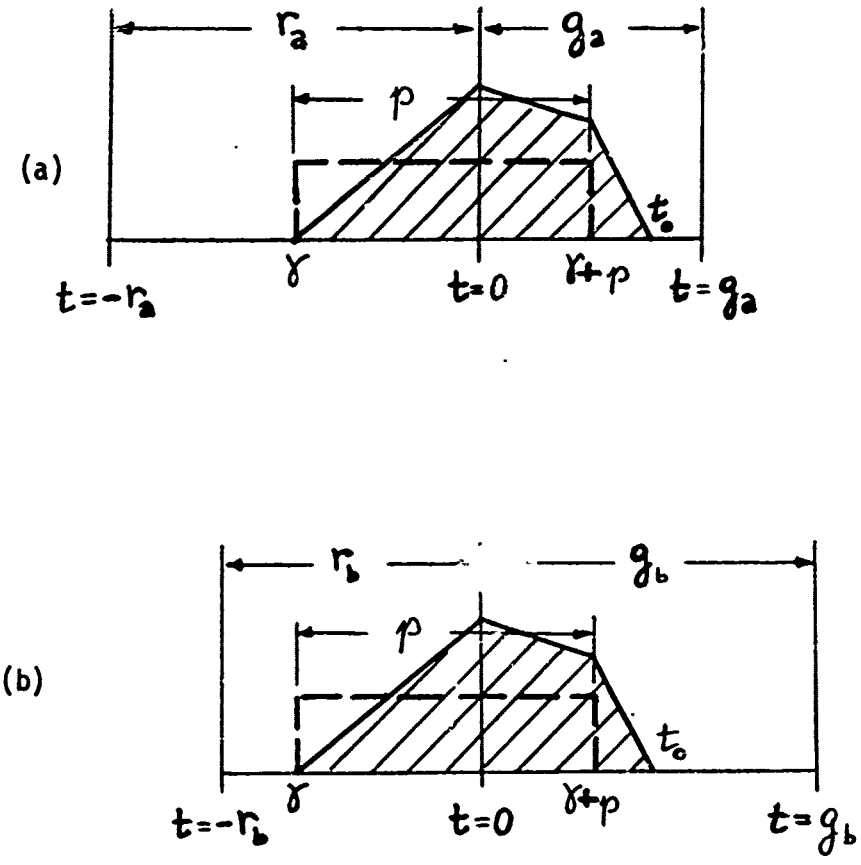


Figure 3.13: Invariability of delay with split.

and,

$$z_{min} = \frac{Z_{min}}{qp} = \frac{1}{2}p(y-1) \tag{3.41}$$

The minimum occurs for $\gamma \geq 0$ and $t_0 \leq g$, i.e., $\gamma + yp \leq g$. Consequently,

$$0 \leq \gamma \leq g-yp \tag{3.42}$$

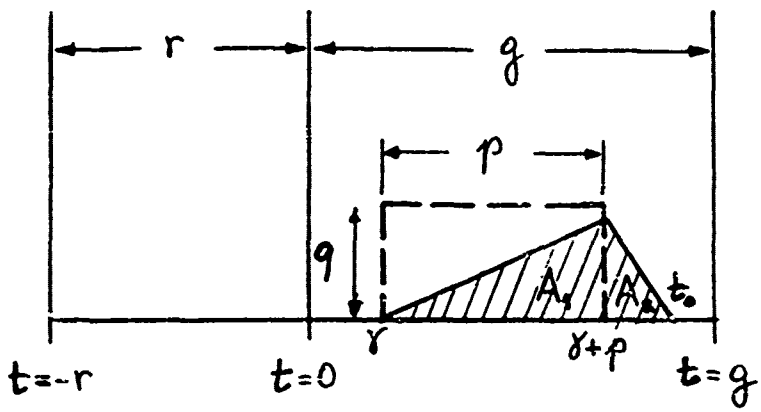


Figure 3.14: Minimal delay for $y > 1$.

4. THE TRAFFIC SIGNAL NETWORK COORDINATION PROBLEM

In this chapter we formulate the traffic signal network coordination problem as follows: given a common cycle time for the network and red and green splits at each node, find a fundamental set of offsets to minimize total delay. The formulation is in terms of mixed-integer linear programming and is followed by an example.

4.1 Piecewise Linearization of the Link Performance Function

The model developed in Section 3 exhibits delay as a nonlinear function of offset, split and cycle time. However, if delay is a function of offset only (split and cycle time are kept fixed in the present formulation) we have a quasi-convex relationship (see Figs. 3.6-3.9). To accommodate mixed-integer linear programming computational techniques it is essential to linearize the objective function (total delay in the network) by piecewise linear approximations. Our convention is to use secant approximating lines rather than tangent lines. They seem to give a fairly accurate approximation, particularly in the region of the minimum, and they are easy to construct. Two examples of such an approximation of delay versus offsets, for two different splits, are shown in Fig. 4.1 and Fig. 4.2.

We shall now transfer the linearized decision function into the constraint set. Suppose we approximate the objective function by n linear segments. We represent each line as $z = z_1(\phi)$ where $z_1(\phi)$ is linear in ϕ. Now consider the set of constraints

$$\begin{aligned} z &\geq z_1(\phi) \\ z &\geq z_2(\phi) \\ &\vdots \\ z &\geq z_n(\phi). \end{aligned} \tag{4.1}$$

They imply that z belongs to a closed convex set $\mathcal{V}$. In our formulation we introduce one linear constraint per link per linear segment which approximates the true objective function. This accounts for the bulk of the constraints of the optimization problem. The number of segments depends upon the desired accuracy of the approximation. Yet one should keep in mind that increasing the number of segments increases the computational burden.

Another property of delay versus offset should be noted. Suppose we have a certain delay z_1 for an offset ϕ_1. If we change this offset by an integral multiple of the cycle time we should obtain the same delay, since delay as a function of offset is periodic with period C. Yet we have modeled delay as a nonperiodic function to maintain convexity. Although the function is not explicitly periodic, the integer variables in the offset

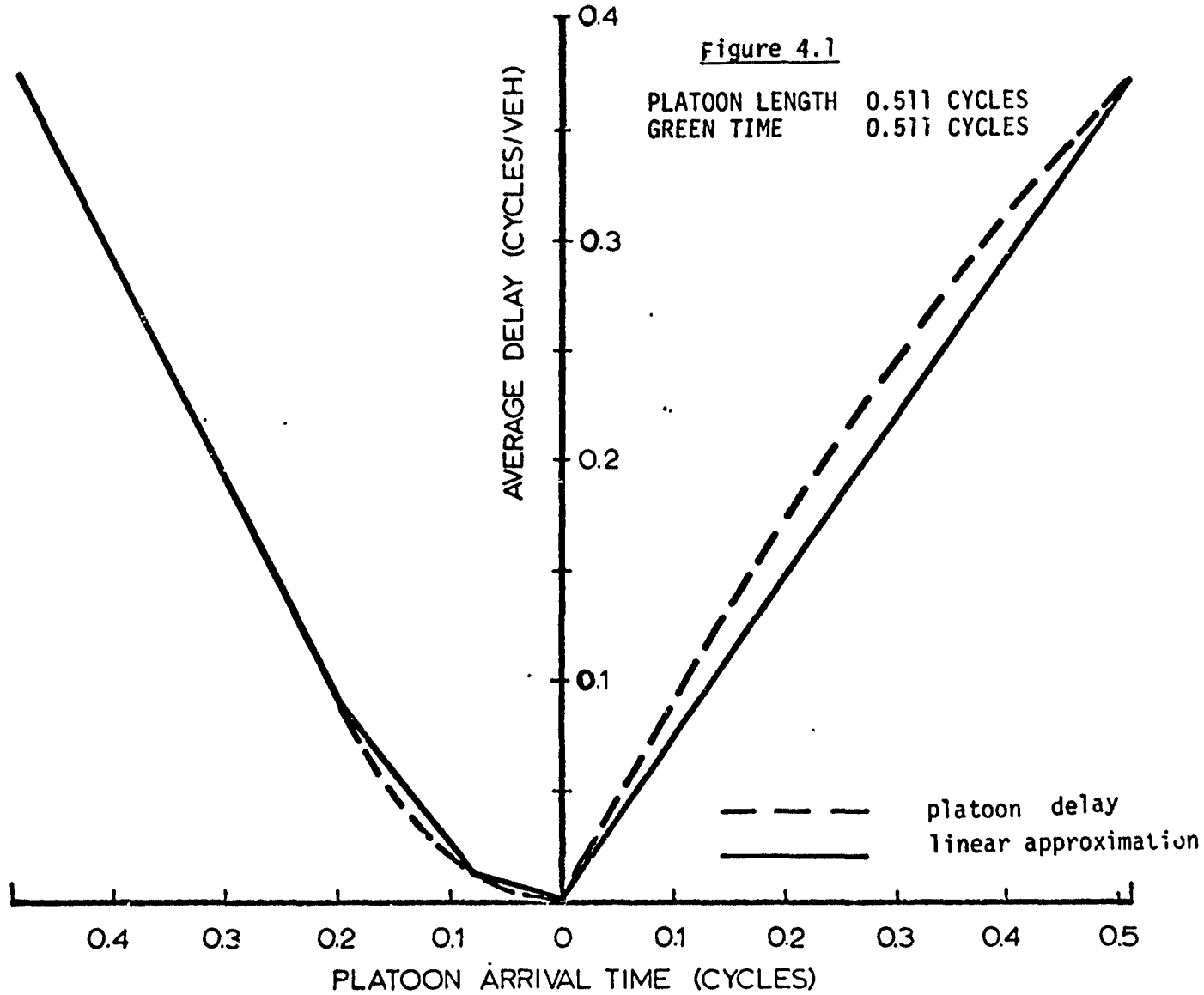
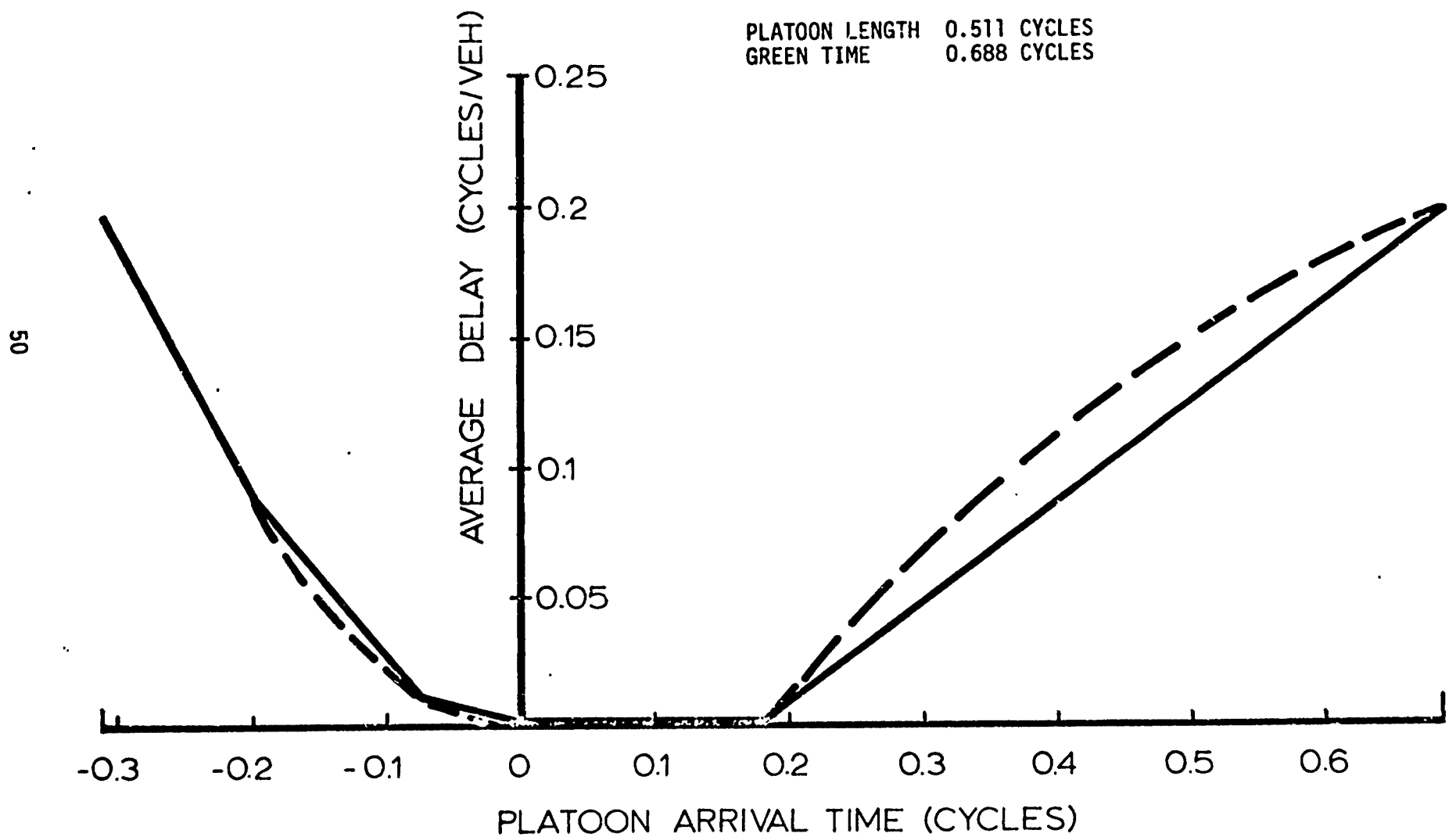


Figure 4.1
PLATOON LENGTH 0.511 CYCLES
GREEN TIME 0.511 CYCLES
AVERAGE DELAY (CYCLES/VEH)
0.4
0.3
0.2
0.1
platoon delay
linear approximation
0.4 0.3 0.2 0.1 0 0.1 0.2 0.3 0.4 0.5
PLATOON ARRIVAL TIME (CYCLES)

Figure 4.2


PLATOON LENGTH 0.511 CYCLES
GREEN TIME 0.688 CYCLES
AVERAGE DELAY (CYCLES/VEH)
0.25
0.2
0.15
0.1
0.05
-0.3
-0.2
-0.1
0
0.1
0.2
0.3
0.4
0.5
0.6
PLATOON ARRIVAL TIME (CYCLES)

loop constraints reflect this property. Suppose we have the piecewise linear convex objective function illustrated in Fig. 4.3. Note that $z(\tau-g) = z(\tau+r)$. We can prove the following theorem:

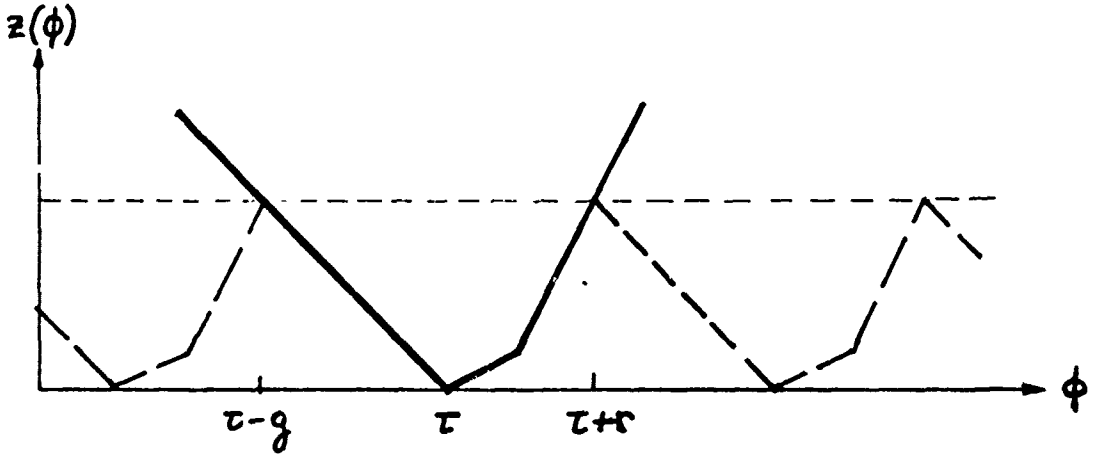


Figure 4.3: Piecewise Linear Convex Objective Function.

Theorem 4.1: Any optimal value of offset ϕ must belong to the interval $[\tau-g, \tau+r]$.

Proof: Suppose not. Let $z^* = z(\phi^*)$ be the associated delay, with $\phi^* \notin [\tau-g, \tau+r]$. There exists an integer m such that $\phi^* + m\,C \in [\tau-g, \tau+r]$ and $z(\phi^* + m\,C) \leq z(\phi^*)$ since, $z(\phi^*) \geq z(\tau+r) = z(\tau-g) \geq z(\phi^* + mC)$. Furthermore, the only physical constraint in which offset appears is the loop constraint, which is not violated by changing any link offset by an integer multiple of cycle time. Thus we maintain feasibility and consistency of all physical constraints and the optimization procedure must produce a value of $z^* = z(\phi^*)$ such that $\phi^* \in [\tau-g, \tau+r]$. Q.E.D.

The next section considers in more detail the network loop constraints.

4.2 The Network Loop Equations

The signal network loop constraints were stated in a general form in (2.5):

$$\sum_{(i,j) \in \ell} \phi_{ij} = n_\ell C \qquad \forall\, \ell \in \mathcal{L} \tag{4.2}$$

Let us consider first loops containing two links in opposing directions between the same pair of nodes. Loops containing two links in the same direction are a trivial case. Referring to Fig. 2.3 we obtain the following constraining equation:

$$\phi_{ij} + \phi_{ji} = n\,C \tag{4.2}$$

Dividing through by C we obtain:

$$\phi'_{ij} + \phi'_{ji} = n \tag{4.3}$$

where $\phi' = \frac{\phi}{C}$ is a dimensionless offset variable (given in fractions of cycle time). The integer n reflects a periodicity in the variable ϕ (or ϕ'), with period C. Since the link performance function as a function of offset has also period C, we may restrict offsets to an interval of length C and obtain bounds on the permissible values for n.

$$\tau_{ij} - g_{ij} \leq \phi_{ij} \leq \tau_{ij} + r_{ij} \tag{4.4}$$

Thus, taking the lower and upper limits, we get according to (4.2)

$$(\tau_{ij} - g_{ij}) + (\tau_{ji} - g_{ji}) \leq n\,C \leq (\tau_{ij} + r_{ij}) + (\tau_{ji} + r_{ji}) \tag{4.5}$$

Defining,

$$\bar{\tau}_{ij} = \tau_{ij} + \tau_{ji} \tag{4.6}$$

and rearranging (4.5) we obtain

$$\bar{\tau}_{ij} \leq n\,C \leq \bar{\tau}_{ij} + 2C \tag{4.7}$$

or

$$(\bar{\tau}_{ij}/C) \leq n \leq (\bar{\tau}_{ij}/C) + 2 \tag{4.8}$$

Let.

$$[x] = \{\text{the greatest integer less than or equal to } x\} \tag{4.9}$$

Since n is restricted to the set of integers, we have,

$$[\bar{\tau}_{ij}/C] + 1 \leq n \leq [\bar{\tau}_{ij}/C] + 2 \text{ if } \bar{\tau}_{ij}/C \text{ is not integer,} \tag{4.10}$$

and,

$$[\bar{\tau}_{ij}/C] \leq n \leq [\bar{\tau}_{ij}/C] + 2 \text{ if } \bar{\tau}_{ij}/C \text{ is integer.} \tag{4.11}$$

If C is allowed to be variable as in Section 6 below

$$C^{min} \leq C \leq C^{max}, \tag{4.12}$$

we obtain for (4.10) and (4.11) respectively:

$$[\bar{\tau}_{ij}/C^{max}] + 1 < n < [\bar{\tau}_{ij}/C^{min}] + 2 \tag{4.13}$$

and,

$$[\bar{\tau}_{ij}/C^{max}] \leq n \leq [\bar{\tau}_{ij}/C^{min}] + 2 \tag{4.14}$$

* * * * * *

For the case where a loop contains more than two links we must extend the notion of offset. So far ϕ_{ij} has been defined as the time measured from start of green at S_i to start of green at S_j along a link (i,j) connecting the two signals. That is, the offset is measured between two different nodes in the same direction (see Fig. 2.1). Now we shall also consider an offset representing the difference in the start of green times at the same node in two different directions. This we shall term an <u>intra-node offset</u>; the time representing the difference between the start of green at S_j in direction (i,j) and the start of green at S_j in direction (j,m). This difference is actually the effective green plus lost time at S_j in direction (i,j), or, $g_{ij} + l_{ij}$ (Fig. 2.2). Distinguishing between link offsets (<u>inter-node offsets</u>) and intra-node offsets, the constraints (2.5) can be rewritten as

$$\sum_{(i,j) \in \ell} (\phi_{ij} + \psi_{ij}) = n_\ell C \qquad \forall \ell \in \mathcal{L} \tag{4.15}$$

where $\psi_{ij} = g_{ij} + l_{ij}$ in this case.

According to (2.6) we can substitute the relation $g_{ij} = C - r_{ij}$ to obtain

$$\sum_{(i,j) \in \ell} (\phi_{ij} - r_{ij} + l_{ij}) = (n_\ell - m) C \tag{4.16}$$

where m is the number of arcs in this particular loop (m > 2).

Letting $n_\ell - m = n$ (integer) we obtain,

$$\sum_{(i,j) \in \ell} (\phi_{ij} - r_{ij} + l_{ij}) = n C \tag{4.17}$$

and, in dimensionless quantities,

$$\sum_{(i,j)\ \epsilon\ \ell} (\phi'_{ij} - r'_{ij} + w\, l_{ij}) = n \tag{4.18}$$

where $w = 1/C$ denotes the signal's frequency [time^{-1}].

To estimate the integer bounds we substitute in (4.16) the lower and upper limits in ϕ, as in (4.4) above:

$$\sum_{(i,j)\ \epsilon\ \ell} (\tau_{ij} - g_{ij} - r_{ij} + l_{ij}) \le n\, C \le \sum_{(i,j)\ \epsilon\ \ell} (\tau_{ij} + r_{ij} - r_{ij} + l_{ij}) \qquad \forall\ \ell \in \mathcal{L} \tag{4.19}$$

Rearranging and substituting

$$T_\ell = \sum_{(i,j)\ \epsilon\ \ell} (\tau_{ij} + l_{ij}) \tag{4.20}$$

we obtain for the bounds,

$$T_\ell - m\, C \le n\, C \le T_\ell \qquad \forall\ \ell \in \mathcal{L} \tag{4.21}$$

or,

$$(T_\ell/C) - m < n < (T_\ell/C) \tag{4.22}$$

For C variable as in (4.12), we obtain,

$$(T_\ell/C^{max}) - m \le n \le (T_\ell/C^{min}) \tag{4.23}$$

or, since n and m are integers,

$$[T_\ell/C^{max}] - m + 1 \le n \le [T_\ell/C^{min}] \text{ if } (T_\ell/C^{max}) \text{ is not integral,} \tag{4.24}$$

and,

$$[T_\ell/C^{max}] - m \le n \le [T_\ell/C^{min}] \text{ if } (T_\ell/C^{max}) \text{ is integral.}$$

It should be noted that not all loop equations will be exactly of the above form. Different pathologies can occur. For example, the loop in Fig. 4.4 contains links in opposing directions. Following (4.15) it gives rise to this equation:

$$\phi_i + g_i + l_i - \phi_j + g_h + l_h + \phi_k + g_k + l_k - \phi_m + g_f + l_f = n_\ell C$$

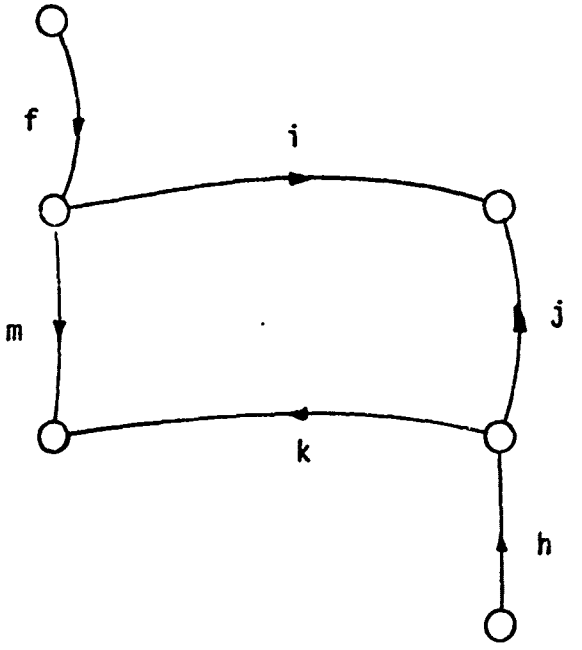


Figure 4.4: Pathology of loop equation.

Essentially as we traverse a loop in time we must add or subtract appropriate time quantities (according to the direction of the links in the loop) to return to the starting point in an integral number of cycle times.

4.3 Mixed-Integer Linear Programming Formulation

Lastly, in a network with |N| nodes and |L| links, exactly |L| - |N| + 1 loop constraint equations are linearly independent [9]. A corollary to this is that |N| - 1 offset variables are independent and they must correspond to links that form a tree subnetwork of the general network. The loops corresponding to the independent set of constraints are called a fundamental set of loops and are denoted by $\mathcal{L}_f$.

The objective in choosing $\mathcal{L}_f$ is related to the computational efficiency of mixed-integer programming. It is desirable to limit the number of integer lattice points of the problem, i.e., maintain close upper and lower bounds on the integer variables. This is achieved by choosing loops that contain the smallest possible number of links. In a planar network this is effected by choosing the meshes of the planar sketch as the fundamental set of loops.

We can now formulate the traffic signal network coordination problem as the following mixed-integer linear programming problem (MILPP):

MILPP: Find ϕ_{ij} to

$$\min \sum_{(i,j) \in L} f_{ij} z_{ij}$$

subject to: $z_{ij} \geq z_{ij}^{k} (\phi_{ij}) \qquad k = 1,2,\ldots,K_{ij}$

$$\forall (i,j) \in L$$

$$\left.\begin{aligned} &\sum_{(i,j) \in \ell} (\phi_{ij} + \psi_{ij}) = n_\ell C \\ &n_\ell^l \leq n_\ell \leq n_\ell^u \\ &n_\ell \text{ integer.} \end{aligned}\right\} \quad \forall \ell \in \mathcal{L}_f$$

K_{ij} denotes the number of linear segments chosen for the (i,j) link performance function, and n_ℓ^l and n_ℓ^u denote respectively the lower and upper bounds on the integer variable.

4.4 Test Network Solution

The following data describe the test network for which a sample problem is actually solved on the IBM 370/165 using IBM's Mathematical Programming Systems (MPSX) package. Figure 4.5 is a sketch of the test network. There are 9 nodes and 24 links of which 16 are internal to the network and 8 are input. The input links although not considered in this problem will be mentioned in later solutions. Tables 4.1 and 4.2 present the data for this and the later examples of the test network.

In this example only offsets are decision variables with green splits and cycle time fixed according to Webster [40]. The critical intersection is No. 7 in Fig. 4.5, yielding an 80 second cycle time:

$$C = \frac{1.5L + 5}{1-Y} = \frac{(1.5 \times 9) + 5}{1-0.767} = 79.5 \text{ sec} \cong 80 \text{ sec}$$

where L = 9 secs. is the total lost time at that intersection (two phases) and

$$y = y_1 + y_2 = \frac{q_1}{s_1} (\text{link } 123) + \frac{q_2}{s_2} (\text{link } 111) = 0.767$$

Tables 4.3, 4.4 and 4.5 present the main output data for this example.

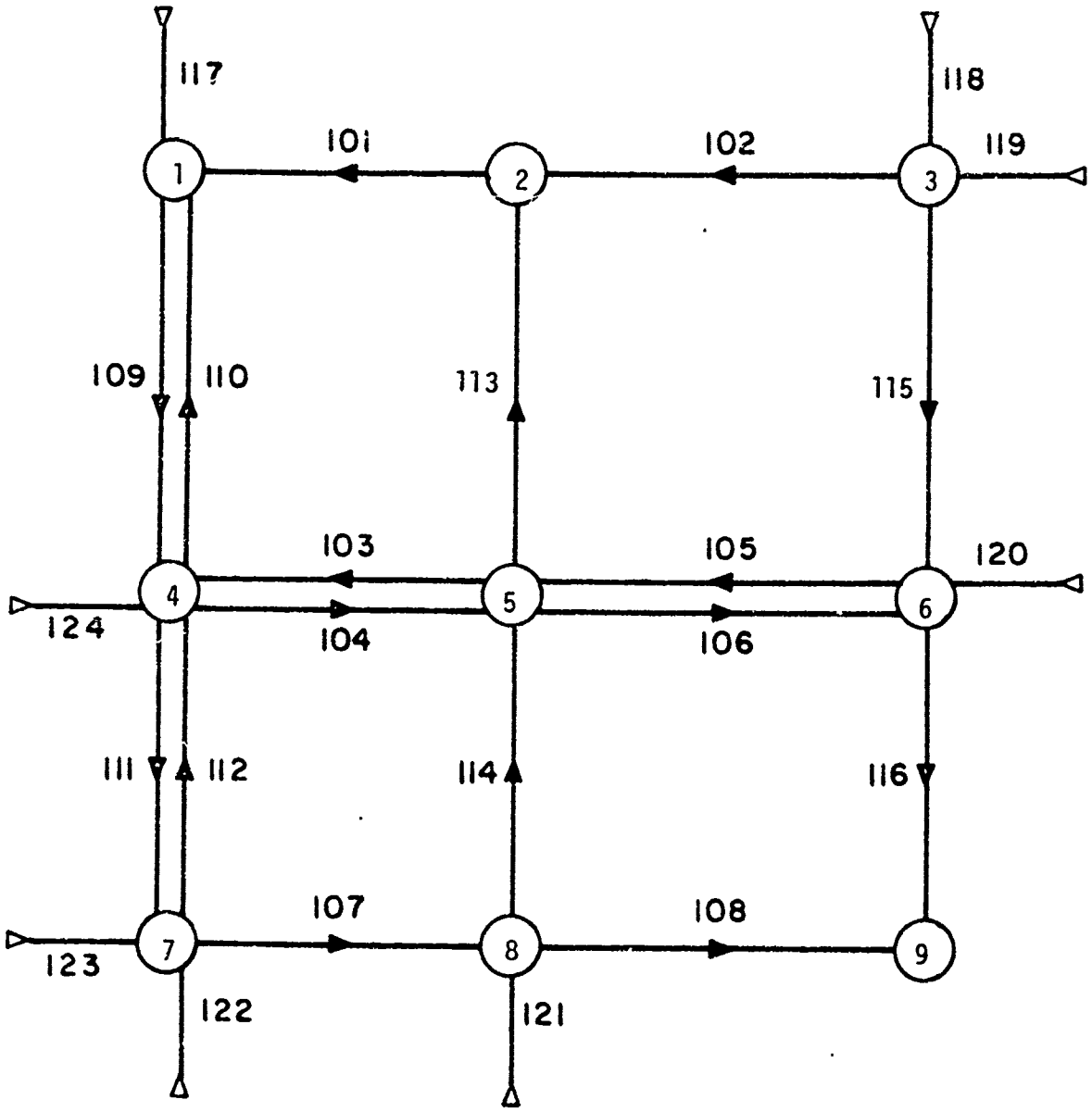


Figure 4.5 - Test network diagram.

Table 4.1

Loops of the Test Network and Their Corresponding Integer Variables

Loop No.	Integer Variable	Link Numbers			
1	N1	109	110		
2	N2	103	104		
3	N3	105	106		
4	N4	111	112		
5	N5	101	109	104	113
6	N6	103	111	102	114
7	N7	115	105	113	102
8	N8	114	106	116	108

Table 4.2

Link Data for Test Network*

Link No.	Link Length (ft)	Velocity (ft/sec)	Travel Time (sec)	Flow (veh/sec)	Sat. Flow (veh/sec)	Platoon Length (cycle)
101	600	32.27	18.60	.152	.835	.335
102	800	32.27	24.79	.152	.835	.344
103	600	32.27	18.60	.152	.835	.418
104	600	41.07	14.61	.22	.835	.407
105	800	32.27	24.79	.152	.835	.463
106	800	41.07	19.48	.22	.835	.444
107	600	41.07	14.61	.25	.600	.515
108	800	41.07	19.48	.25	.600	.560
109	1000	35.20	28.41	.175	.500	.701
110	1000	41.07	24.35	.11	.500	.608
111	550	35.20	15.63	.175	.500	.532
112	550	41.07	13.39	.11	.500	.425
113	1000	41.07	24.35	.12	.360	.595
114	550	41.07	13.39	.12	.360	.414
115	1000	35.20	28.41	.175	.500	.701
116	550	35.20	15.63	.175	.500	.503
Input Links						
117				.175	.500	
118				.175	.500	
119				.152	.835	
120				.152	.835	
121				.12	.360	
122				.11	.500	
123				.25	.600	
124				.22	.835	

*(a) Lost time for each link, l = 4.5 secs.

(b) Platoon length on input links, p = 1 cycle (i.e., flow is assumed to be continuous, though random fluctuations will be taken into account in subsequent formulations).

Table 4.3

Test Network Results and Statistics*

I.

Rows	161
Columns	97
Variables	258
Integer Variables	8
Non-zero Elements	709
Density	1.70

II.

	Time (min)	Iteration No.	Node No.	Functional Value
Continuous Optimum	.03	76	1	43.3461
First Integer Solution	.06	124	18	62.6695
Optimal Integer Solution	.07	132	21	60.8741
Optimality Proved	.21	305	57	
Time of Search	.21	305	57	

III. Number of Integer Variables Not Integer at Continuous Optimum = 8
Number of Integer Solutions Found = 2
Branches Abandoned While Computing = 50

*The number of rows and columns refer to the linear programming coefficient matrix. The number of continuous variables includes one variable for each column plus one slack variable for each row. Thus the total number of variables equals the number of rows plus columns. Integer variables correspond to the constraints associated with each network loop. The density is the percentage of non-zero elements to the total number of elements in the linear programming coefficient matrix.

In solving the MILP, the integer variables are first treated as continuous and the resulting LP is solved yielding the continuous optimum which is a lower bound for all possible feasible integer solutions. Yet this continuous solution is an infeasible integer solution. At each level (Fig. 4.6) an integer variable is chosen with a non-integer value and two problems are considered. One is restricting the variable to be greater or equal to the smallest integer greater than this continuous value, the other is restricting the integer variable to be less than or equal to the largest integer less than or equal to the continuous value. Then each of these problems is solved again as an LP and the next node considered is the one that produced the smallest increase in the objective function. This process continues until we finally obtain the first integer solution. The program then proceeds to follow all other possible paths perhaps finding improved integer solutions. Once all feasible possibilities are considered optimality is proved and the last integer solution is chosen as the optimal solution [7]. The graphical representation of this process is called a Branch and Bound Diagram (Fig. 4.6).

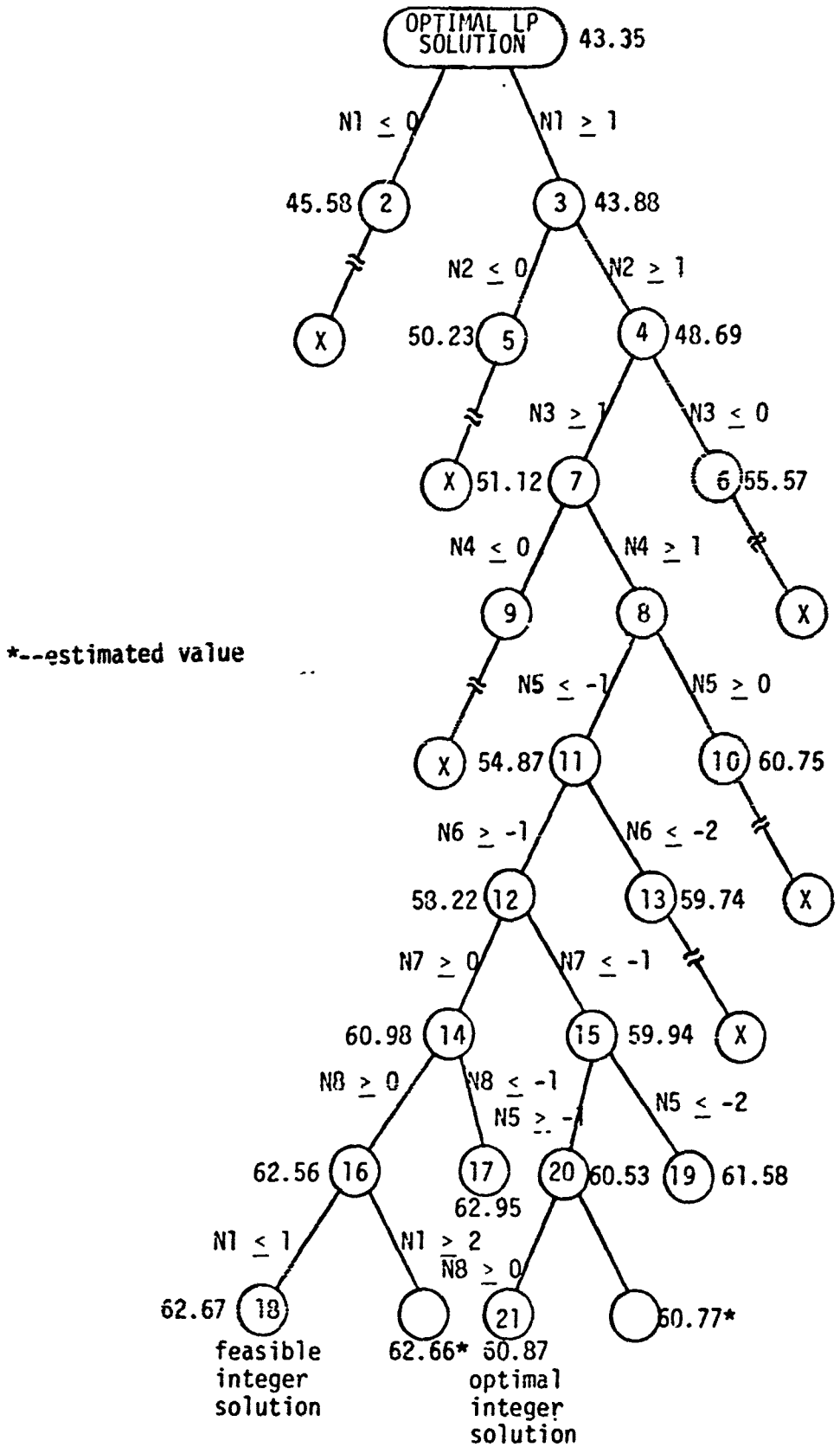
Table 4.4

Integer Nodes

B-B Node	18	21
Functional Value	62.6696	60.8741
Integer Variable	Integer	Integer
N1	1	1
N2	1	1
N3	1	1
N4	1	1
N5	-1	-1
N6	-1	-1
N7	0	-1
N8	0	0

Figure 5.6 is the Branch and Bound Diagram corresponding to the MILP output of the test network. The nodes are numbered according to the order in which they are reached. Nodes with x indicate fathoming*, while proving optimality. Specific nodes and values were omitted from the sketch for simplicity. The branching is indicated along the line segments connecting two nodes. The value of the objective function is given beside each node while an asterisk denotes an estimated value.

*(i.e., further searching along these branches is abandoned, since solution is inferior to the one already established).



*--estimated value

Figure 4.6: Branch and Bound diagram. Splits and cycle time predetermined, offsets variable.

Table 4.5

Result of Computations*
(C = 80 sec; cycle time and
green times fixed according to Webster)

Link No.	Offset sec (cycle)	Green Times sec (cycle)	Delay	SDF	Degree of Saturation
101	21.6 (.27)	24.0 (.30)	1.36	0	.607
102	27.2 (.34)	24.8 (.31)	1.36	0	.587
103	64.0 (.80)	30.4 (.38)	36.0	0	.479
104	16.0 (.20)	31.2 (.39)	.72	0	.676
105	47.2 (.59)	31.2 (.39)	11.28	0	.467
106	32.8 (.41)	32.8 (.31)	6.64	0	.850
107	16.8 (.21)	39.2 (.49)	1.04	1.422	.850
108	25.6 (.30)	38.4 (.48)	3.2	1.574	.868
109	26.4 (.33)	40.8 (.51)	10.88	0	.686
110	53.6 (.67)	46.4 (.58)	14.8	0	.603
111	32.8 (.41)	32.0 (.40)	9.6	1.891	.875
112	47.2 (.59)	40.8 (.51)	25.92	0	.686
113	16.8 (.21)	45.6 (.57)	5.36	0	.585
114	47.2 (.59)	40.0 (.50)	30.64	.027	.667
115	46.4 (.58)	38.4 (.48)	8.88	.450	.729
116	24.0 (.30)	32.0 (.40)	4.16	1.891	.875
Input Links					
117		46.4 (.58)	12.16	0	.603
118		46.4 (.58)	12.16	0	.603
119		24.0 (.30)	24.56	0	.607
120		32.8 (.41)	18.64	0	.444
121		31.2 (.39)	22.16	2.107	.855
122		32.0 (.40)	19.76	0	.550
123		38.4 (.48)	18.64	1.574	.868
124		30.4 (.38)	21.52	0	.693

*The delay due to the Saturation Deterrence Function (SDF) has no effect on the optimization procedure in this case since it is invariant with offsets, and splits and cycle time are kept fixed. The same is true for the delay values due to the LPF on the input links. Both factors become important when splits and cycle time are decision variables too (see Sections 5 and 6).

5. VARIABLE SPLIT FORMULATION

In this chapter we add the split at each node of the network as a decision variable to our formulation of the traffic signal network optimization problem. For simplicity we consider only r_{ij} since $g_{ij} = C - r_{ij}$. Now the objective function of delay previously dependent on offset alone is a function of red times also. Two effects are added. First, we must consider the input links of the network. Second, a phenomenon not considered before has an effect on the split decision--the stochastic behavior of traffic flow. This is modeled by a saturation deterrence function (SDF). Finally we give the complete formulation followed by an example.

5.1 Link Performance Function - Approximations by Planes

To obtain piecewise linear relations in the (γ,r,z) space we have to fit planes by selecting triplets of points. A plane

$$z = k_1 + k_2 \gamma + k_3 r \tag{5.1}$$

can be formed from the points $P_i(\gamma_i,r_i,z_i)$, $i = 1,2,3$ by the following simple relationships from geometry:

$$\left.\begin{aligned} k_1 &= z_1 - k_2\gamma_1 - k_3 r_1 \\ k_2 &= [(z_2 - z_1)(r_3 - r_1) - (z_3 - z_1)(r_2 - r_1)]/\alpha \\ k_3 &= [(z_3 - z_1)(\gamma_2 - \gamma_1) - (z_2 - z_1)(\gamma_3 - \gamma_1)]/\alpha \\ \alpha &= [(\gamma_2 - \gamma_1)(r_3 - r_1) - (\gamma_3 - \gamma_1)(r_2 - r_1)] \end{aligned}\right\} \tag{5.2}$$

Referring to Fig. 5.1 we define five planes $\mathcal{A}$ by the following triplets:

$$\mathcal{A}_I : \{P_1,P_2,P_3\}$$

$$\mathcal{A}_{II} : \{P_2,P_3P_4\}$$

$$\mathcal{A}_{III} : \{P_i \mid z_i = 0\}$$

$$\mathcal{A}_{IV} : \{P_4,P_5,P_6\}$$

$$\mathcal{A}_V : \{P_2,P_4,P_6\}$$

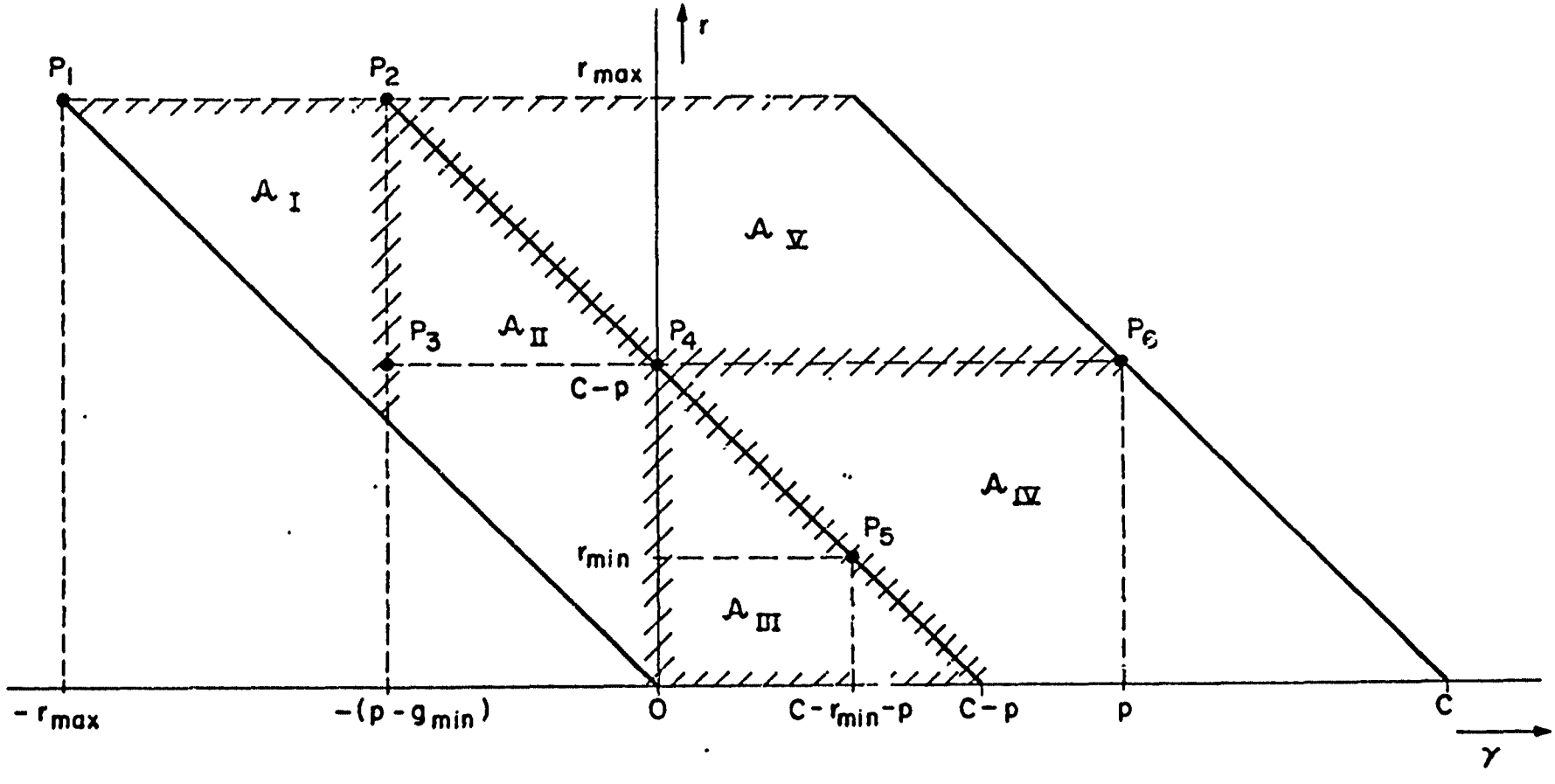


Figure 5.1: Plane approximations to Link Performance Function.

The coordinates γ_i, r_i, z_i are determined as follows:

$$P_1 : \gamma_1 = -r_{max}$$

$$r_1 = r_{max}$$

$$z_1 = z_{max}(\text{at } r_1)$$

$$P_2 : \gamma_2 = -(p - g_{min}) = C - r_{max} - p$$

$$r_2 = r_{max}$$

$$z_2 = z_{min}(\text{at } r_2)$$

$$P_3 : \gamma_3 = \gamma_2$$

$$r_3 = r_4$$

$$z_3 = z_2$$

$$P_4 : \gamma_4 = 0$$

$$r_4 = C - p$$

$$z_4 = 0$$

$$P_5 : \gamma_5 = C - r_{min} - p$$

$$r_5 = r_{min}$$

$$z_5 = 0$$

$$P_6 : r_6 = p$$

$$r_6 = r_4$$

$$z_6 = z_{max}(\text{at } r_4)$$

r_{max} is determined by capacity requirements as follows:

$$qp \leq gs \tag{5.3}$$

or,

$$qp = g_{min}\, s = (C - r_{max})s.$$

Consequently,

$$r_{max} = C - yp \tag{5.4}$$

z_{max} at a certain split r is given by (3.27) and (3.36) as follows:

$$z_{max} = \frac{p}{2}(y - 1) + r \quad \text{if} \quad p(1 - y) \leq r$$

$$z_{max} = \frac{r^2}{2p} \cdot \frac{1}{1 - y} \quad \text{if} \quad p(1 - y) > r$$

z_{min} at a certain split r is given by (3.24):

$$z_{min} = \frac{(p - g)^2}{2p} \cdot \frac{1}{1 - y} = \frac{(p - C + r)^2}{2p} \cdot \frac{1}{1 - y}$$

Planes $\mathcal{A}_I$ and $\mathcal{A}_{II}$ are parallel to the r-axis, according to Theorem 3.5. Therefore:

$$k_3^I = k_3^{II} = 0$$

and we can determine the planes by two points $P_i(r_i, z_i)$ only, as follows (see Fig. 5.1):

$$\mathcal{A}_I : \{P_1, P_2\}$$

$$\mathcal{A}_{II} : \{P_3, P_4\}.$$

The plane equations in this case are of the form:

$$z = k_1 + k_2\gamma$$

Finally, to obtain the piecewise linear suface as a function of offsets we have to substitute for γ relationship (2.2) as follows:

$$\underset{\substack{\text{arrival}\\\text{time}}}{\gamma} = \underset{\substack{\text{travel}\\\text{time}}}{\tau} - \underset{\text{offset}}{\phi}$$

The planes $\mathcal{A}_I,\ldots,\mathcal{A}_V$ enclose a convex subspace that constitutes a piecewise linear approximation to the three-dimensional surface $z(\gamma,r)$. The points defining the planes have been chosen to approximate most closely the bottom part of the valley-like enclosure, i.e., where the optimal solutions are eventually expected. A typical set of $z(\gamma,r)$ that were calculated for a rectangular-shapped platoon and their approximation via the planes described above, are shown in Fig. 5.2. Note that the line connecting (P_4,P_2) has a parabolic shape, according to equation (3.24). If a higher accuracy is warranted, this line can be approximated more closely by a series of secant planes, rather than the single plane $\mathcal{A}_V$.

5.2 Adjusted Traffic Flow Model

In Section 3 the platoon characteristics were calculated according to the green splits at the intersection, which were fixed and precalculated according to Webster's formulas. Since in the current context the upstream green time is itself a decision variable, a modified procedure has to be employed. Given the cycle time, we first determine nominal splits at each node, e.g., by Webster's method. These splits are used to determine the platoon lengths according to Section 3.2. Though the final splits might be slightly different than those assumed for calulating p, we believe that this procedure provides a satisfactory approach. An alternative approach would be to conduct an iterative process in which platoon legths are upoated according to the current values for the splits. The result would be either convergence, or a "limit cycle" such as experienced by nonlinear control systems. In the first case no distinguishable difference will exist between split values at successive iterations. The second case may result in oscillations of the split values with small amplitudes around an average value. Further computational experimentation is required to evaluate this approach.

5.3 Modeling the Network Input Links

In the current formulation splits are considered as decision variables at all nodes of the network. Input links are a prime determinant of the

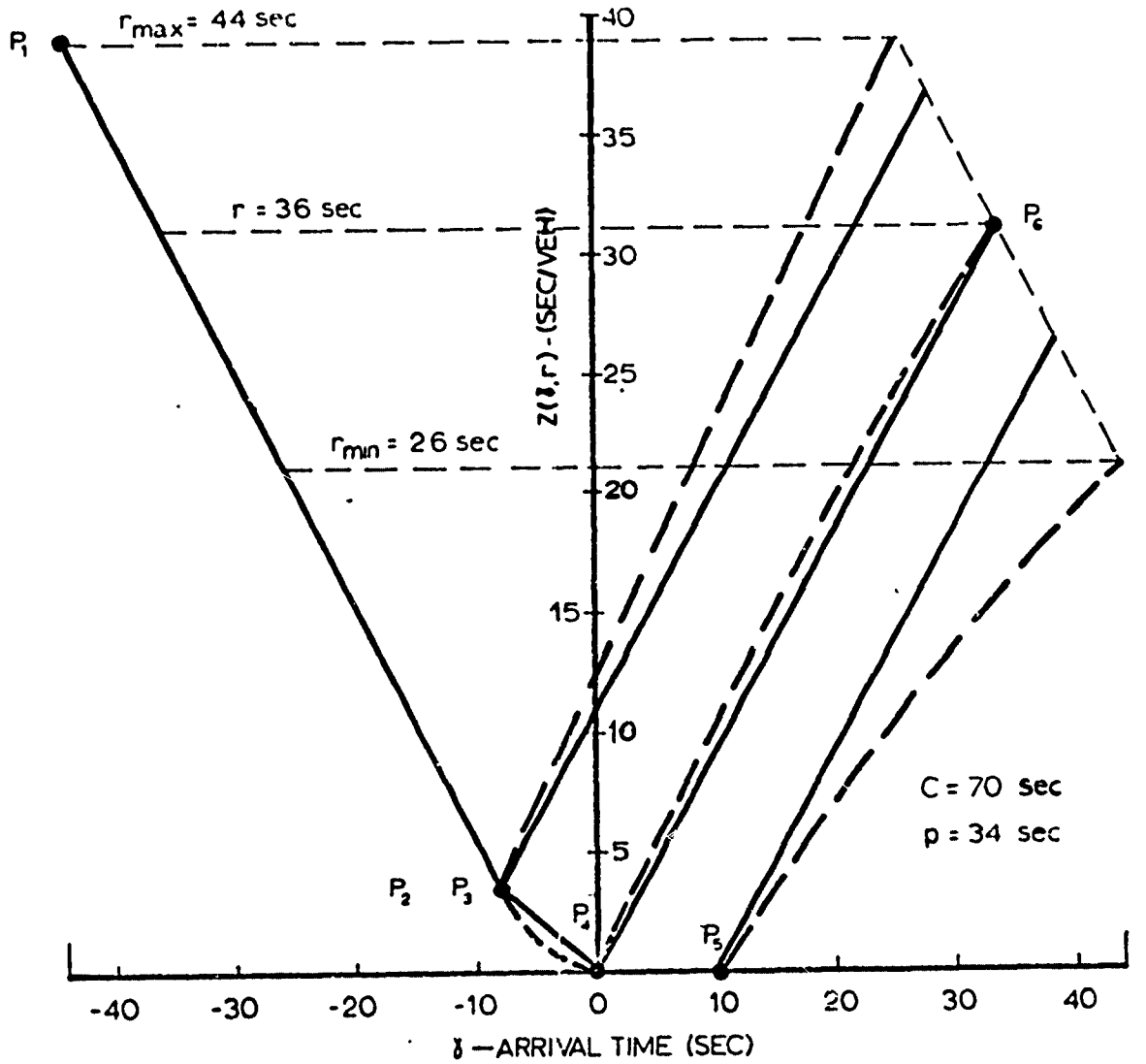


Figure 5.2: Projections of (z, t, r) space.

splits at the signals on the boundary of the network and must be included in the program. Traffic on these links is assumed to be independent of any neighboring signals and therefore will be modeled as a continuous flow, i.e., with p = 1. Though, stochastic effects will be included, as explained below in Section 5.4.

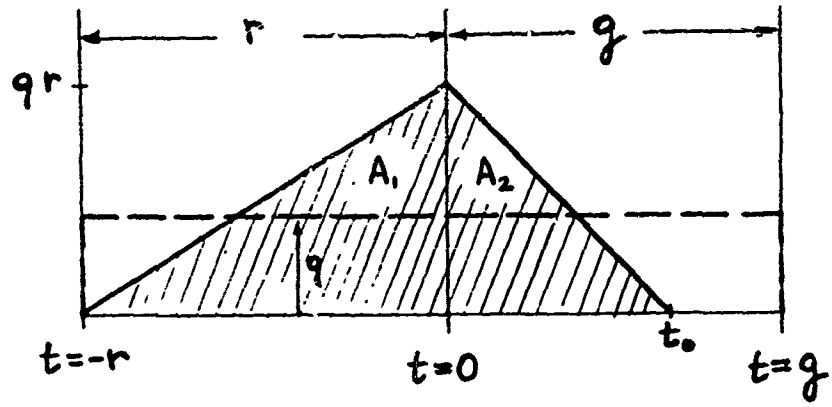


Figure 5.3: Queue evolvement on input links (p = r + g = C).

Referring to Fig. 5.3, for p = r + g = C we have for the clearance time t_0:

$$qr + qt_0 - st_0 = 0.$$

Hence,

$$t_0 = \frac{qr}{s - q} = r \frac{y}{1 - y}. \tag{5.5}$$

The total delay is a sum of two areas:

$$A_1 = \frac{1}{2} qr^2; \; A_2 = \frac{1}{2} qrt_0 = \frac{1}{2} \frac{qr^2 y}{1 - y}$$

$$Z = A_1 + A_2 = \frac{1}{2} \frac{qr^2}{1 - y} , \tag{5.6}$$

and the average delay is,

$$z = \frac{Z}{q(r + g)} = \frac{1}{2} \frac{r^2}{(r + g)(1 - y)} = \frac{1}{2} \frac{r^2}{C(1 - y)} \tag{5.7}$$

Equation (5.7) is equivalent to the deterministic delay component in Webster's delay formula[40].

5.4 Stochastic Effects - The Saturation Deterrence Function

An important physical phenomenon is missing in the performance functions of most signal optimization procedures. At flows that are close to capacity but still, on the average, below it, occasional fluctuations in the size of the platoon can lead to temporary overflow queues that seriously degrade performance. This is a stochastic phenomenon which has a consequence that, as average flow approaches capacity, average delay increases, gradually at first, and then very rapidly.

A representation of this effect is needed to prevent green time from approaching its lower bound too closely. It is, of course, possible to put a sizeable constraint on minimum green time but such an approach misses the main idea of an optimization, which is to tradeoff delay in one part of the system against that in another. The effect is particularly essential if cycle length is a variable to be determined by the optimization process as will be shown later in Section 6. This is because of the tradeoff between capacity loss at short cycles and the inherently large delays of long cycles.

Wormleighton [43] has studied the effect in some detail motivated by experience with the Toronto traffic control system. Following his model, assume

1) Arriving vehicles come in platoons or other periodic function of time with an average arrival rate at time t of q(t).

2) Arrivals are a non-homogeneous Poisson process. Thus the number of vehicles in (t,t + C) is a random variable having a Poisson distribution with mean

$$A_C = \int_t^{t+C} q(u)\, du \tag{5.8}$$

for any t.

3) The service provided by the green time is deterministic with rate s(vehicles/sec) up to gs(vehicles/cycle).

If the state of the intersection is examined at the end of green, a bulk service queuing model is defined. Let

$Q(0)$ = number of vehicles in queue at the start of red (end of green) --overflow queue

$A_c = fC$ = average number of vehicles arriving in a cycle

$S = gs$ = number of vehicles that can be served in a green time.

$x = A_c/S = fC/gs$ = utilization factor (degree of saturation).

Wormleighton finds the generating function of Q(0) and calculates E{Q(0)} as a function of S and x over $S \in [5,55]$ and $x \in [.2,.975]$. Within these ranges E{Q(0)} varies from essentially zero to 18 vehicles.

We wish to understand how the presence of overflow queues affects delays as a function of offset. Insight into the situation is obtained by examining the deterministic case. A theorem is proved to show that under certain circumstances delay can be broken into two components, one depending only on the overflow queue and the other being the delay that would be incurred in the absence of the overflow queue. This motivates the introduction of the overflow queue effect into the link performance function in a simple way.

Consider the sketch of queue vs. time during a cycle shown in Fig. 5.4.

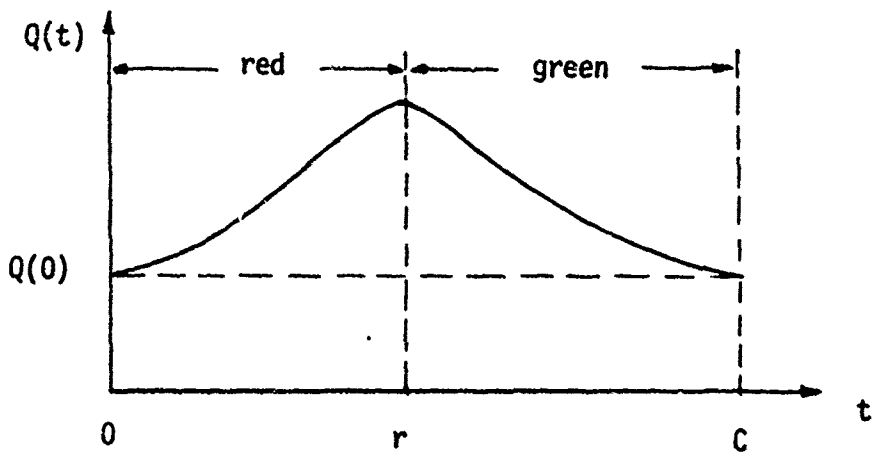


Figure 5.4: Queue vs. time during a cycle (with overflow queue).

Let

$q(t)$ = arrival rate at t (veh/sec)

$A_C = \int_0^C q(t)\,dt$ = total arrivals in cycle

$Q(t)$ = number in queue at time t

$Q(0)$ = overflow queue at start of red

$Q_0(t)$ = number in queue at t if overflow queue had been $Q(0) = 0$.

$Z = \int_0^C Q(t)\,dt$ = aggregate delay over cycle (veh/sec)

$Z_0 = \int_0^C Q_0(t)\,dt$ = aggregate delay if overflow queue had been $Q(0) = 0$(veh/sec)

s = service rate during green while vehicles are present in queue(veh/sec)

It will be assumed that $q(t) \leq s$, i.e., the instantaneous arrival rate never exceeds the instantaneous service rate.

Theorem 5.1:

If $A_C = gs$, i.e., the number of arrivals equals the maximum number of vehicles that can be served, then

(a) the overflow queue is preserved $Q(C) = Q(0)$.

(b) the aggregate delay is composed of a term equal to the wait of the overflow queue and a term equal to the delay in the absence of the overflow queue:

$$Z = Q(0)\ C + Z_0 \tag{5.9}$$

Proof: See Fig. 5.4. Because $A_c = gs$, if all of the green is used for serving vehicles, the queue returns to its starting point and $Q(C) = Q(0)$. If not all the green is used, the ending queue will be higher. In either case

$$Q(C) \geq Q(0) \tag{5.10}$$

Because $q(t) \leq s$, the queue never increases during green and the minimal queue during green is $Q(C)$, i.e.,

$$Q(t) \geq Q(C) \quad (t \text{ in green}) \tag{5.11}$$

Therefore, if $Q(C) > 0$, some queue is present throughout green and all green time is used for serving. In this case, by previous argument, the queue returns to its starting point

$$Q(C) = Q(0) \tag{5.12}$$

On the other hand if $Q(C) = 0$, $Q(0) = 0$ by (5.10), and (5.12) is trivially true. Thus in any case (5.12) holds, proving (a). Notice that $Q(t)$ never falls below $Q(0)$.

Next observe that the order of service does not affect the number in queue and so does not affect the aggregate delays. Suppose, therefore, that the overflow queue vehicles are served last. Then they are, in fact, never served and the arriving vehicles are served just as if $Q(0) = 0$. The delay of the vehicles arriving during the cycle is Z_0. The delay of the overflow queue vehicles is $CQ(0)$. The total delay is $Z = Z_0 + CQ(0)$ thereby proving (b).

We use the above result to argue that at high flow when $A_c \simeq gs$, it is reasonable to separate the total delay into two components, one being the result of the presence of an average overflow queue, the other being the normal delay of vehicles in average flow, in the absence of an overflow queue.

It has not been established that this is exactly true in the stochastic case and so the term involving the overflow queue should be regarded as a deterrence function which involves increasing penalties to performance as capacity is approached. Thus we use Wormleighton's results to determine the saturation deterrence function (SDF), which enters additively into the objective function. In terms of delay per unit of time, the SDF component is simply $Q(0) \equiv Q$. The overflow queue Q as a function of the number of release

Table 5.1

EXPECTED OVERFLOW QUEUE, Q

(Adpated from Ref. [43])

S = gs, No. of release points per cycle	x, degree of saturation						
	0.20	0.40	0.60	0.80	0.90	0.95	0.975
5	0.00	0.02	0.20	1.15	3.50	8.41	18.36
15	0.00	0.00	0.04	0.70	2.81	7.61	17.50
25	0.00	0.00	0.01	0.47	2.41	7.08	16.91
35		0.00	0.00	0.34	2.11	6.68	16.45
45			0.00	0.23	1.88	6.34	16.05
55					1.68	6.02	15.67

points and the degree of saturation is listed in Table 5.1.

5.5 Piecewise Linearization of the SDF

We must be able to represent the delay of the SDF in a form amenable to mixed integer linear programming techniques. Although there is no simple analytic expression for Q, it is convex in r. We seek ways to linearize this convex function. Essentially we will treat this delay as we did with the link performance function, first using secant approximating lines, then transferring them into the constraint set with corresponding delays contributing additively to the original objective function (in Section 4.3).

Three lines will be used to estimate this delay as a function of red times (see Fig. 5.5). For any particular line we restrict r to be such that the degree of saturation remains below 0.95.

Table 5.2

Example of Expected Overflow Queue vs. Red Times

C = 60 sec

s = .833 veh/sec S = gs $x = \frac{fC}{gs}$

f = .278 veh/sec

$\frac{r}{C}$	$\frac{g}{C}$	S	x	Q
.05	.95	47.5	.351	0.0
.10	.90	45.0	.370	0.0
.15	.85	42.5	.392	0.0
.20	.80	40.0	.417	0.0
.25	.75	37.5	.444	0.0
.30	.70	35.0	.476	0.0
.35	.65	32.5	.513	0.0
.40	.60	30.0	.556	0.0
.45	.55	27.5	.606	.01
.50	.50	25.0	.667	.16
.55	.45	22.5	.741	.38
.60	.40	20.0	.833	1.2
.65	.35	17.0	.952	7.48

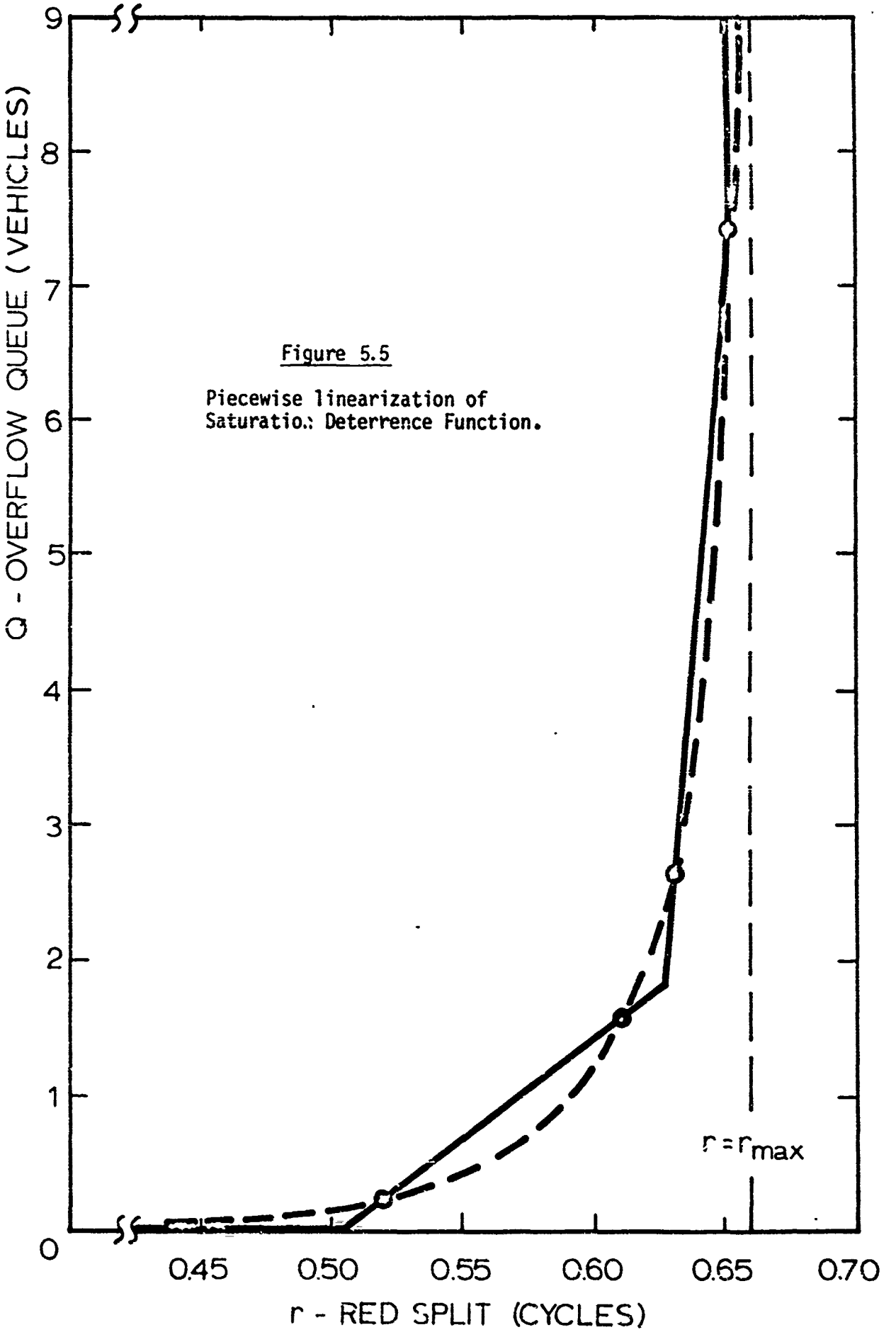


Figure 5.5

Piecewise linearization of
Saturation Deterrence Function.

Our capacity constraint reads

$$g_{ij}s_{ij} \leq f_{ij}C$$

or in terms of red times

$$r_{ij} \leq C(1 - f_{ij}/s_{ij}).$$

This upper limit on red splits is obtained by considering

$$0.95g_{ij}s_{ij} \leq f_{ij}C.$$

Note that $f_{ij}C/g_{ij}s_{ij}$ is the degree of saturation x. The other two lines are determined by the following procedure: We are given flows and saturation flows for every link. The cycle time C has been determined according to Webster. For the first line we choose the degree of saturation x = .95 and x = .90 which determines the corresponding S values (S = gs = f C/x, i.e., the number of release points). Then we obtain the corresponding Q values from the table either directly or by interpolation if necessary for each link. The second line is determined by x = .85 and x = .70, yielding again directly or by interpolation the corresponding Q values. An example to the final approximation to the nonlinear curve is given in Fig. 5.5. The data for the original curve is given in Table 5.2.

These lines appear in the constraint set. We require the SDF delay, Q, to lie above each line and represent the total delay from this effect as the sum of individual delays over all the links of the network. It should be mentioned that the first line, $r_{ij} \leq C(1 - f_{ij}/.95\ s_{ij})$ is not handled as a constraint but rather as an upper bound to the red split variable. Computationally this is more efficient.

5.6 Formulation as a Mixed-Integer Linear Programming Problem

Now we are able to present the complete MILP formulation with offsets and splits decision variables. We denote input links by the ordered pair (I,j) where j is a boundary node of the network, to distinguish from internal links denoted by (i,j).

MILP: Find ϕ_{ij}, r_{ij} to

$$\min \sum_{\substack{(i,j)\ \epsilon\ L \\ (I,j)\ \epsilon\ L}} (f_{ij} z_{ij} + Q_{ij})$$

subject to:

$$z_{ij} \geq z^k(\phi_{ij}, r_{ij}) \qquad k = 1,\ldots,K(\text{LPF--internal links})$$

$$z_{Ij} \geq z^{k'}(r_{Ij}) \qquad k' = 1,\ldots,K(\text{LPF--input links})$$

$$Q_{ij} \geq Q^{l'}(r_{ij}) \qquad l' = 1,\ldots,L(\text{SDF--internal links})$$

$$Q_{Ij} \geq Q^{l''}(r_{Ij}) \qquad l'' = 1,\ldots,L(\text{SDF--input links})$$

$$\sum_{(i,j)\ \epsilon\ \ell} (\phi_{ij} + \psi_{ij}) = n_\ell C \qquad \ell \in \mathcal{L}_f$$

$$r_{ij} - l_j = r_{kj} - l_j \qquad j \in N;\ i,k \in P_j$$

$$r_{ij} - (l_j + a_{ij}) = g_{mj} + l_j \qquad j \in N;\ i \in P_j;\ m \in \bar{P}_j$$

$$r_{ij}^{\min} \leq r_{ij} \leq 1 - f_{ij}/0.95\, s_{ij} \qquad \forall (i,j) \in L$$

$$n_\ell^l \leq n_\ell \leq n_\ell^u \qquad (n_\ell \text{ integer})$$

ϕ_{ij} unrestricted in sign.

5.7 Test Network Solution

Our test network of Fig. 4.5 is now solved for offsets and splits variable. The cycle time is again determined according to Wester (C = 80sec). The results are given in Tables 5.3,5.4, and 5.5. The Branch and Bound diagram corresponding to the search conducted by the program in pursuit of the optimal integer values is given in Fig. 5.6. (For explanation of terms see Section 4.4).

Table 5.3

Test Network Results and Statistics

I.	Rows	176
	Columns	97
	Variables	273
	Integer Variables	8
	Non-zero Elements	763
	Density	1.58

	Time (min)	Iteration No.	Node No.	Functional Value
II. Continuous Optimum	.04	160	1	42.6181
First Integer Solution	.08	236	18	61.4830
Second Integer Solution	.15	250	21	59.6584
Optimal Integer Solution	.20	384	47	59.6269
Optimality Proved	.37	576	67	
Time of Search	.37	576	67	

III. Number of Integer Variables not Integer at Continuous Optimum = 8
Number of Integer Solutions Found = 3
Branches Abandoned While Computing = 58

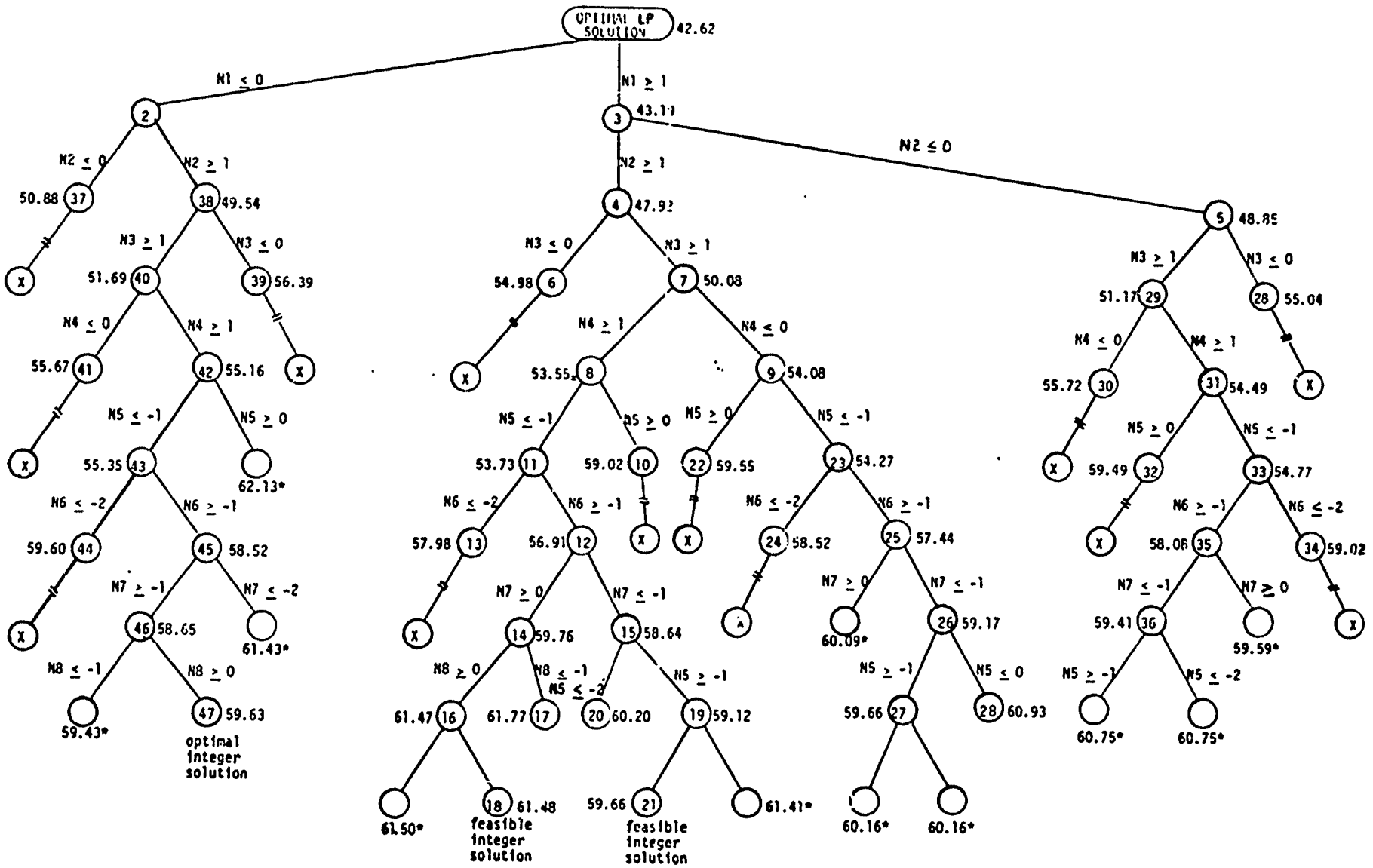


Figure 5.6 Branch and Bound diagram. Cycle time predetermined, offsets and splits variable.

Table 5.4

Integer Nodes

B-B Node	18	21	27
Functional Value	61.4830	59.6584	59.6269
Integer Variable	Integer	Integer	Integer
N1	1	1	0
N2	1	1	1
N3	1	1	1
N4	1	1	1
N5	-1	-1	-1
N6	-1	-1	-1
N7	0	-1	-1
N8	0	0	0

(The B-B nodes refer to Fig. 5.6).

Table 5.5

Results of Computations
(C = 80 sec. according to Webster)

Link No.	Offset sec (cycle)	Green Times sec (cycle)	Delay	Delay Due to SDF	Degree of Saturation
101	24.0 (.30)	21.6 (.27)	2.72	0	.674
102	28.8 (.36)	23.2 (.29)	2.08	0	.628
103	62.4 (.78)	29.6 (.37)	34.64	0	.492
104	17.6 (.22)	29.6 (.37)	1.36	0	.712
105	46.4 (.58)	29.6 (.37)	10.64	0	.492
106	33.6 (.42)	30.4 (.38)	7.2	0	.693
107	14.4 (.18)	40.8 (.51)	.08	1.067	.817
108	24.8 (.31)	40.0 (.50)	2.56	1.339	.833
109	18.4 (.23)	39.2 (.49)	13.36	.279	.714
110	-18.4 (-.23)	49.6 (.62)	13.52	0	.355
111	35.2 (.44)	33.6 (.42)	11.92	1.505	.833
112	44.8 (.56)	39.2 (.49)	23.6	0	.449
113	24.0 (.30)	47.2 (.59)	0.0	0	.565
114	44.0 (.55)	49.8 (.51)	27.36	0	.654
115	41.6 (.52)	40.0 (.50)	8.32	0	.700
116	24.8 (.31)	31.2 (.39)	4.48	2.035	.897
Input Links					
117		49.6 (.62)	9.92	0	.565
118		49.6 (.62)	9.92	0	.565
119		21.6 (.27)	26.4	0	.674
120		30.4 (.38)	20.08	0	.479
121		29.6 (.37)	23.04	2.410	.901
122		33.6 (.42)	18.64	0	.524
123		37.6 (.47)	19.36	1.784	.887
124		29.6 (.37)	21.92	0	.712

Section 5.2 indicates that a cause-effect relationship exists among the signal splits which are decision variables of the program, and the platoon model of traffic flow, which is an input parameter to the optimization procedure. Iterating through the sequence: platoon → splits → platoon → splits, the optimized splits were used to recalculate platoons. The results are given in Table 5.6. The initially calculated splits are satisfactory, in the sense that no appreciable differences in the resulting platoons are produced by iterating.

Table 5.6

Iteration on Platoon Length

(C = 80 sec)

Link No.	p_1	g_1	p_2	g_2
101	.335	.27	.312	.27
102	.344	.29	.279	.28
103	.418	.37	.398	.37
104	.407	.37	.398	.37
105	.463	.37	.434	.37
106	.444	.38	.423	.37
107	.515	.51	.497	.47
108	.560	.50	.582	.47
109	.701	.49	.745	.49
110	.608	.62	.588	.62
111	.532	.42	.515	.42
112	.425	.49	.442	.49
113	.595	.59	.618	.61
114	.414	.51	.393	.51
115	.701	.60	.746	.51
116	.503	.39	.530	.41

This can be easily interpreted with reference to Fig. 5.7. Considering for example a two-phase intersection we have:

$$g_1 + g_2 + 2l = C$$

split g varies from a minimum of g^{min} (determined by capacity requirements) and g^{max} (determined by capacity requirements in the conflicting phase). Therefore,

$$g_1^{max} = C - 2l - g_2^{min}$$

The combination of the SDF values for both approaches will have the parabolic shape shown in Fig. 5.7. Consequently any choice of offsets will most likely lie in the neighborhood of the minmum, which is very closely approximated by Webster's split-decision formula.

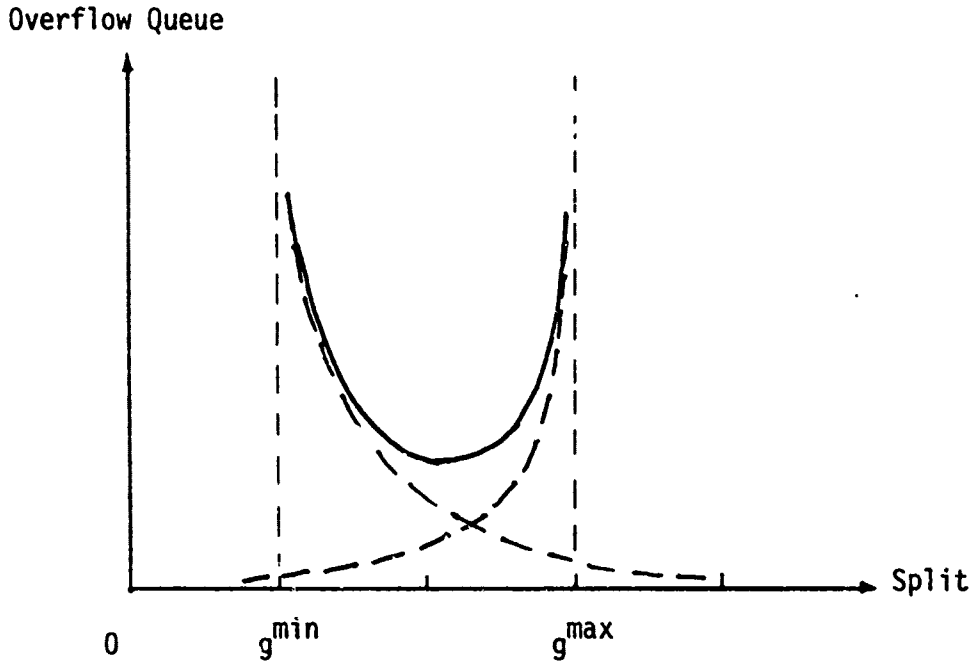


Fig. 5.7: Effect on overflow queue of split value.

6. The Network Synchronization Problem

In the preceding sections we have formulated the traffic signal network problem as minimizing delay due to the deterministic link performance function (LPF) and the stochastic saturation deterrence function (SDF) subject to loop and other physical constraints. We first considered the offset optimization problem (the network coordination problem) and then added the splits at each intersection as decision variables. How we shall let the cycle time, common for all the intersections of the network, be a decision variable too. This will be termed as the network synchronization problem.

6.1 The Optimal Cycle Time in a Network

Experience of researchers and practitioners in the urban traffic control field has shown that the cycle time may well be the most important among the decision variable [3,17,38]. Not only does it tie together the offsets in conjuction with integer variables but it also provides for the necessary capacity to serve the traffic demand at each intersection. The net serving capability of an intersection is linearly proportional to the sum of the effective green times over all the phases of the intersection, i.e., capacity is proportional to

$$\sum_i g_{ij} = C - L \qquad \forall_j \tag{6.1}$$

Since L, the total lost time, is a fixed quantity at a particular intersection, the net capacity increases with cycle time. For example, for a two phase intersection with a 9 second total lost time (4.5 seconds per phase), capacity increases by varying cycle time from 40 seconds to 120 seconds by the difference between 9/40 and 9/120, i.e. by 15%. This meager difference in net capacity may have a decisive impact on delay, particularly at high degrees of saturation. Similar characteristics are well known in many other queueing processes [29]. On the one hand an increase in cycle time is usually followed by an increase in red times on individual phases and consequently by an increase in the average waiting time as expressed by the LPF. On the other hand, a decrease in cycle time reduces capacity, thus precipitating additional delay due to the randomness in arrivals of vehicles. This effect may become particularly severe when flow approaches saturation and is modeled by the SDF.

The interplay between the LPF and the SDF in the objective function for variable cycle time was studied with respect to the test network introduced in Section 4.4. The MILPP formulation of Section 5.7 was run for various fixed cycle times in the range of 40 seconds to 120 seconds to optimize offsets and splits. The resultant values for the objective function, broken down into its two components, are given in Table 6.1. A graphical illustration of the relationships is presented in Fig. 6.1. It becomes evident that the optimal cycle time for the network constitutes a least-cost equilibrium point between delays caused by deterministic effects and delays contributed by stochastic effects. While the first delays usually increase with cycle length (though at a decreasing rate), the latter decrease with it owing to the decrease in the

Table 6.1

Variation of Delay with Cycle Time for Test Network of Section 4.4[veh x sec/sec]

CYCLE (sec)	LPF	SDF	TOTAL DELAY
50	27.5	43.6	71.1
55	31.3	29.5	60.8
63.7	37.7	15.4	53.1
70	42.5	13.1	55.6
80	49.2	10.4	59.6
90	53.5	9.4	62.9
103	57.1	8.7	65.8
120	59.8	8.4	68.2

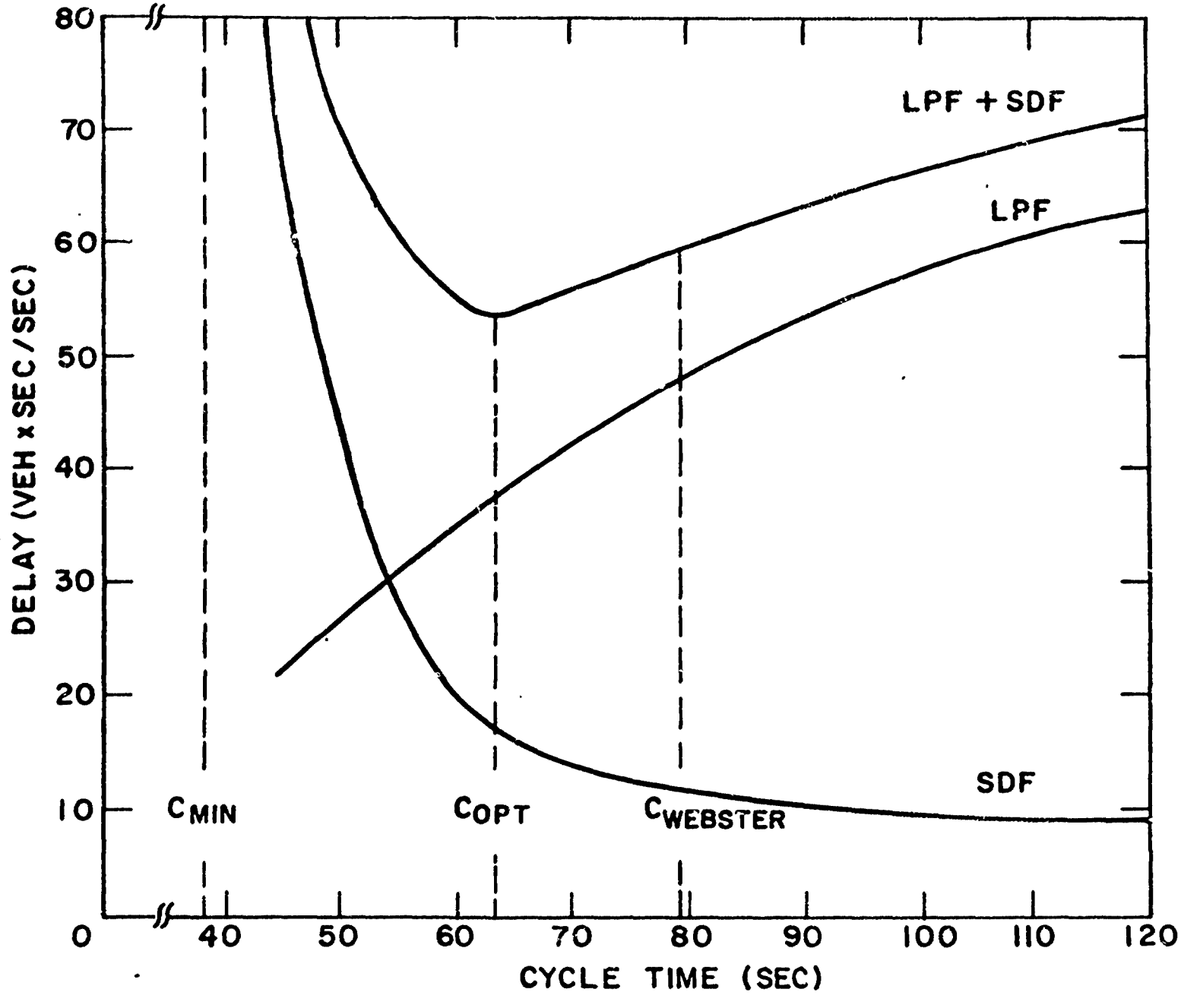


Figure 6.1 Variation of delay with cycle time for test network of Section 4.4.

degree of saturation (the load factor). They approach asymptotically from above the minimal cycle time for the network, which is the theoretical minimal cycle time for the most heavily loaded intersection if all flows were strictly deterministic. These characteristics are in complete analogy with the behaviour of delay with respect to cycle time at a single intersection and were studied by Wardrop[39], Webster[40] and other researchers [26,32]. However, the implications regarding signal settings in a network are different and a single intersection analysis would virtually never give the optimum settings for the network.

6.2 Variable Cycle Formulation

As shown above, to optimize signal settings in a network, any optimization procedure must make a judicious choice of cycle time. To introduce the cycle time as a decision variable our formulation of the network problem has to be modified. It will be observed that by a change of variable the cycle time enters linearly into the LPF and only with slight modifications into the SDF. Examination of the loop constraints with C variable will provide the motivation for this change of variable. The input links will not be affected. Special consideration must be given to the transformed objective function and its linearized representation.

For variable cycle time our objectives become twofold. First we must examine our model of delay and the physical constraints with this added variable. Then we must insure that the optimization procedure can handle these changes. Examining the loop constraints(2.5) we observe that

$$\sum_{(i,j)\ \epsilon\ \ell} \phi_{ij} = n_\ell C$$

for variable cycle time there is a nonlinearity in the term $n_\ell C$. Common means of handling such a nonlinearity as separation of variables or Taylor approximation do not seem to be of use in this case since n_ℓ is an integer variable. Yet by dividing each term of this equation by C we can remove this nonlinearity, introducing terms like ϕ/C, r/C and $1/C$ which are also nonlinear. This suggests a change of variable as follows

$$\phi' = \phi/C,\ r' = r/C,\ w = 1/C \qquad (6.2)$$

i.e., we transform variables from seconds to dimensionless fractions of a cycle time. From this point on we drop all primes on variables, understanding that they are measured in fractions of a cycle time.

So far platoon lengths p have been determined in seconds. Following the above transformation, p will be measured in fractions of a cycle too. We assume that the ratio of platoon length to cycle length (in seconds) remains invariant as we allow the cycle time to vary. This assumption is reasonable since the platoon length on any link (i,j) is considered to be proportional to the upstream green which varies linearly with cycle time. Thus the physical constraints remain linear wiht the above transformation. Now we must examine the constraints corresponding to the different components of the delay function.

The link performance function has been expressed in terms of arrival times γ and splits, according to (2.2)

arrival time		offset		travel time
γ	=	ϕ	-	τ

Transforming arrival time to fractions of a cycle we obtain

$$\gamma' = \phi' - \tau w \tag{6.3}$$

so the reciprocal cycle time enters linearly into the constraints representing the LPF. The LPF for the input links is independent of C and requires no such transformations.

6.3 SDF - Approximation by Planes

The cycle time will now be introduced into the SDF through its reciprocal $w = 1/C$. Referring to Table 5.1 we notice that the expected overflow queue Q is a function of the degree of saturation x and the number of release points in the cycle, S (i.e., the capacity of the approach). By definition we have,

$$S = sgC = s\frac{1 - r}{w} \tag{6.4}$$

(g and r are given in fractions of cycle time)

$$x = \frac{f}{sg} = \frac{f}{s(1 - r)} \tag{6.5}$$

Solving for r and w we obtain

$$r = 1 - \frac{f}{sx} \tag{6.6}$$

$$w = \frac{f}{Sx} \tag{6.7}$$

We may now transform the triplet of points (Q,q,x) into (Q,r,w) which coincides with our decision space.

Figure 6.2 shows a typical relationship between Q,r, and w which suggests planar approximations as before. The first plane, an upper bound on red times, confining the degree of saturation to values below 0.95, will be

$$r \leq 1 - \frac{f}{0.95s} \tag{6.8}$$

as in the case with fixed cycle time (Section 5.5). The other two planes will be determined by the triplets of points:

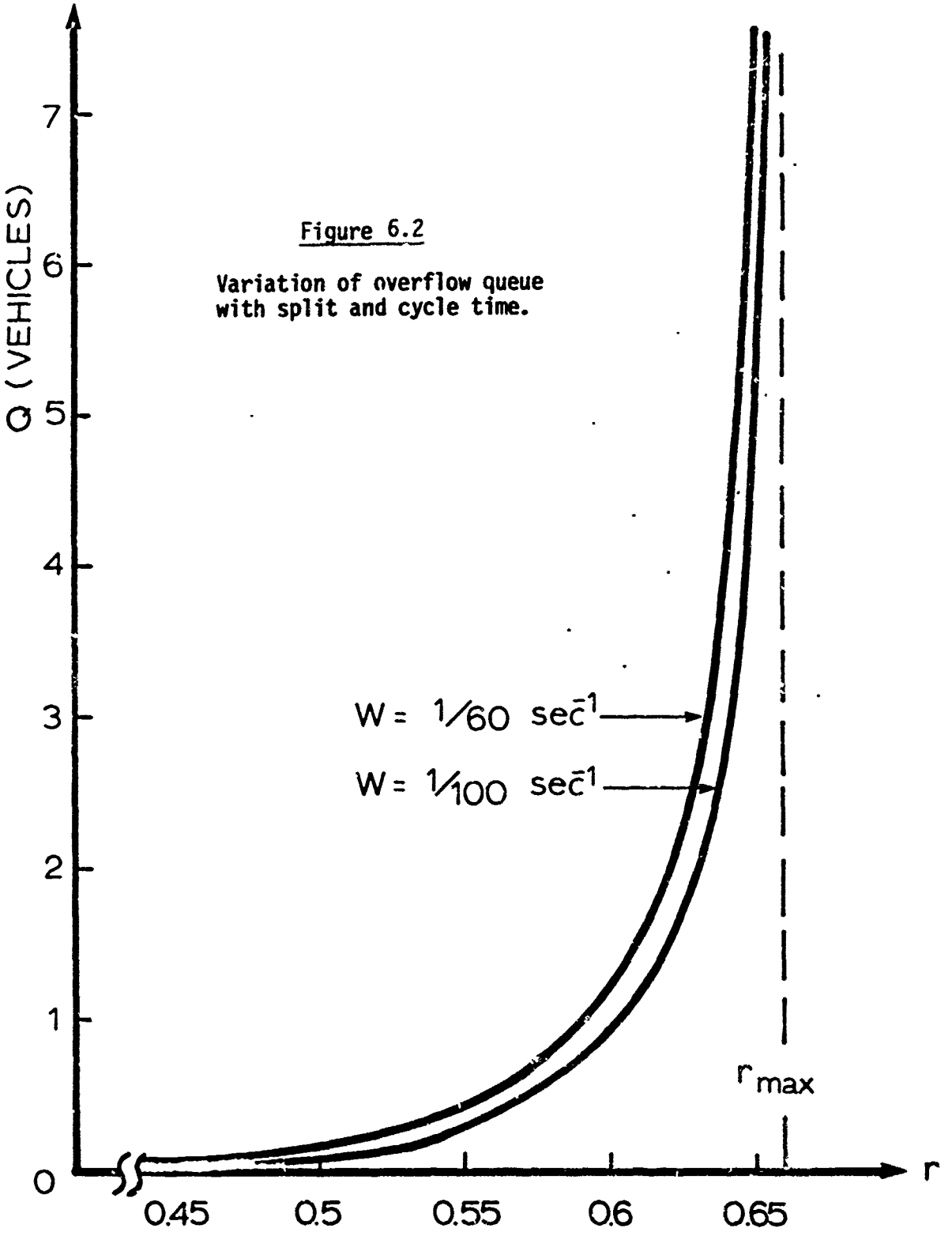


Figure 6.2

Variation of overflow queue with split and cycle time.

	Q	S	x
	7.61	15	0.95
Plane I:	2.81	15	0.90
	2.11	35	0.90
	1.76	15	0.85
Plane II:	1.23	35	0.85
	0.37	15	0.70

The values for Plane II were obtained by interpolation from the values given in Table 5.1. Note that the values for Q change more pronouncedly on the horizontal axis in Table 5.1 than in the vertical axis. This means that Q is more sensitive to variations in split than in cycle time, as evidenced also by the plot in Fig. 6.2. As a result the planes representing the SDF (appearing as contraints in the MILP formulation)

$$Q + ar + bw \geq 0 \tag{6.9}$$

show that the coefficients a are considerably larger in magnitude than the coefficients b.

6.4 Revised Objective Function

Introducing C as a variable has the most significant effect on the objective function. Prior to this change our objective function was the simple linear expression

$$\sum_{(i,j)} (f_{ij} z_{ij} + Q_{ij}), \quad \text{where } z_{ij}$$

is in seconds. Incorporating C as a decision variable, thus transforming variables to dimensionless quantities, introduces a change in the objective function. Now we must consider z_{ij} in fractions of a cycle. We define $z' = z/C$, or

$$z = Cz' = \frac{z'}{w} \tag{6.10}$$

Thus our objective function now reads

$$\min \sum_{(i,j)} (f_{ij} \frac{z'_{ij}}{w} + Q_{ij}).$$

The transformation of z to dimensionless z', provides us with a LPF that is invariant with cycle time. No change occurs in the variable Q_{ij} due to the SDF.

We cannot proceed in a straightforward manner since z/w is nonlinear (we drop again primes). Although it can be shown that z/w is quasi-convex, which may lend itself to gradient techniques, it is not convex. In order to maintain the MILP formulation, we introduce again piecewise planar approximations of z/w. That is, we approximate

$$d_{ij} = \frac{z_{ij}}{w} \tag{6.11}$$

by a set of planes forming a convex region. As before, the planes will be transferred into the constraint set of the objective function, which will now read

$$\min \sum_{(i,j)} (d_{ij} + Q_{ij}).$$

Care must be taken in approximating a non-convex surface by a set of planes forming a convex surface. The difficulty is that we may cut away large portions of the feasible region. Thus we want to approximate the surface as closely as possible in the domain in which we expect the solution to lie and only exclude extreme portions of the feasible region where we are quite sure will be unlikely for a solution to occur. This is exactly the case in the approximation to the LPF, where there is a non-convex portion corresponding to very high green splits.

The following three planes are chosen to approximate the d = z/w surface for internal links of the network. Each of the planes is determined by three points P (see Fig. 6.3).

		z	w	d = z/w
Plane I:	P_1 :	0.15	0.018	0.15/0.018
	P_2 :	0.15	0.0.4	0.15/0.014
	P_3 :	0.00	1/120	0.00
Plane II:	P_1 :	0.15	0.01	0.15/0.01
	P_2 :	0.15	0.014	0.15/0.014
	P_3 :	0.00	1/120	0.00
Plane III:	P_1 :	0.15	0.018	0.15/0.018
	P_2 :	0.00	0.00	0.00
	P_3 :	0.00	0.00	0.00

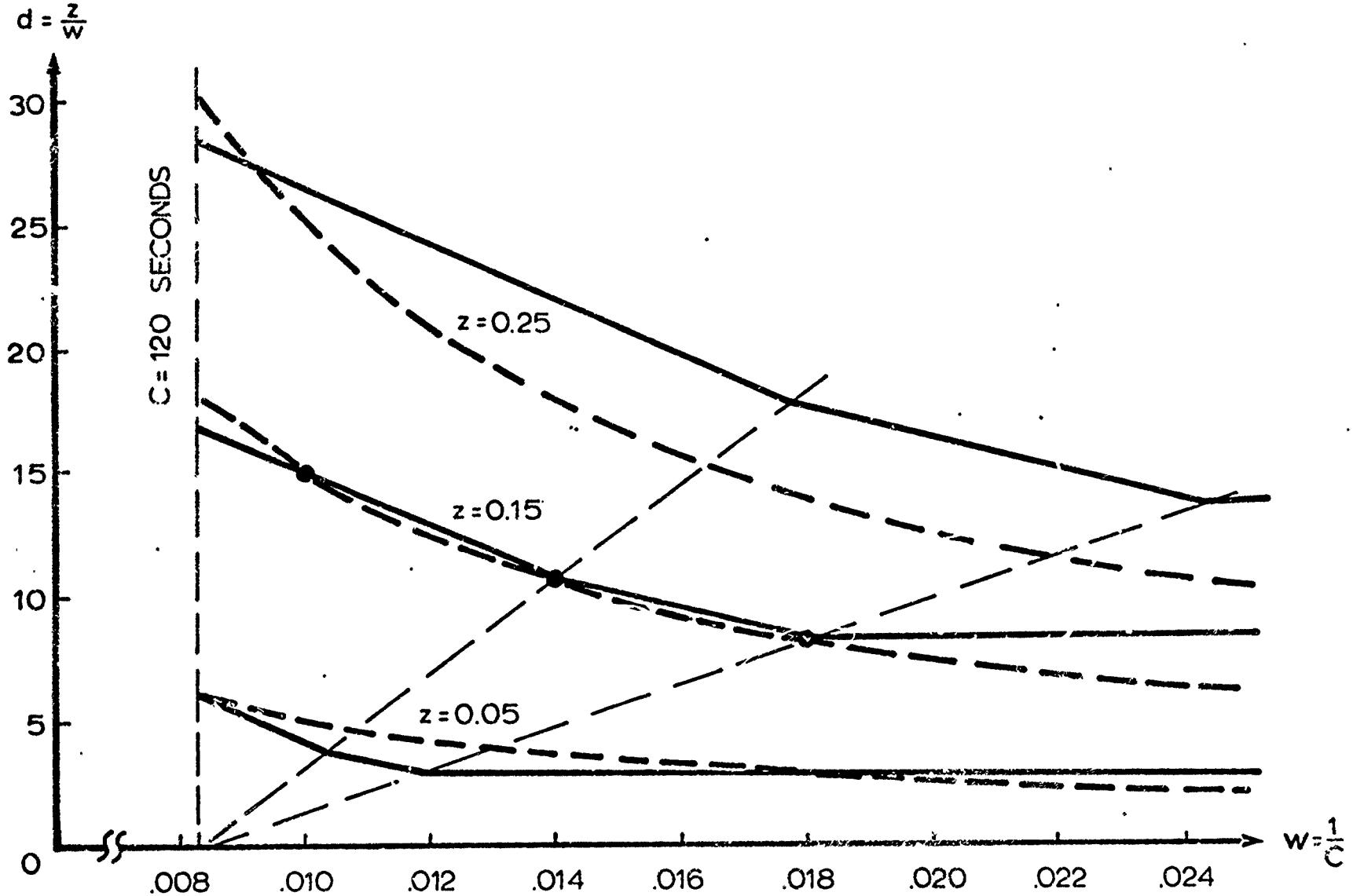


Figure 6.3: Linear approximations to d=z/w space.

The planes have been constructed taking into account that the lower limit on w is 1/120 (corresponding to an upper limit on cycle time of 120 seconds). z = 0.15 has been chosen as the pivot value, representing a typical average delay per car figure (in fractions of cycle time).

The planes approximating the LPF on input links to the network have been constructed similarly, with a pivot value of z = 0.25. The reason is that platoons on these links are assumed to have full cycle length, i.e., flow is uniformly distributed over time. Thus the average delay for vehicles on those links is expected to be higher than on the internal links of the network.

6.5 Mixed-Integer Linear Programming Formulation

The complete optimization problem including simultaneously offsets, splits and frequency (cycle time) as decision variables is formulated as follows:

MILPP: Find ϕ_{ij}, r_{ij}, w to

$$\min \sum_{(i,j) \in L} (f_{ij} d_{ij} + Q_{ij})$$

subject to:

$$z_{ij} \geq z^k(\phi_{ij}, r_{ij}, w) \qquad k = 1,\ldots,K(\text{LPF})$$

$$Q_{ij} \geq Q^l(r_{ij}, w) \qquad l = 1,\ldots,L(\text{SDF})$$

$$d_{ij} \geq d^m(z_{ij}, w) \qquad m = 1,\ldots,M(\text{representation of } z/w)$$

$$\sum_{(i,j) \in \mathcal{L}} (\phi_{ij} + r_{ij}) = n_{\mathcal{L}} \qquad \mathcal{L} \in \mathcal{L}_f$$

$$r_{ij} - l_{ij} w = r_{kj} - l_{kj} w \qquad j \in N;\ i,k \in P_j$$

$$r_{ij} - (l_{ij} + a_{ij}) w = g_{mj} + l_{mj} \qquad j \in N;\ i \in P_j;\ j \in \bar{P}_j$$

$$r^{min} \leq r_{ij} \leq 1 - f_{ij}/0.95\, s_{ij} \qquad (i,j) \in L$$

$$w^{min} \leq w \leq w^{max}$$

$$n_\ell^l \leq n_\ell \leq n_\ell^u \qquad (n_\ell \text{ integer})$$

ϕ_{ij} unrestricted in sign.

6.6 Test Network Solution

The test network of Fig. 4.5 is now solved for offsets, splits and cycle time variable simultaneously. The results are given in Tables 6.2,6.3, and 6.4 below. The Branch and Bound diagram followed by the MILP program in search for the optimal integer set is illustrated in Fig. 6.4. As seen in Table 6.3 the optimal solution in this case is a considerable improvement over those obtained previously (see Section 4.4 and Section 5.7), due to the decisive role played by the cycle time in the total optimization. By considering all the decision variables simultaneously (objective function = 53.80), network performance was improved by 11.6% with respect to settings obtained when the decision variables were considered sequentially (objective function = 60.87).

Table 6.2

Test Network Results and Statistics

I.	Rows	248
	Columns	121
	Variables	369
	Integer Variables	8
	Elements	1027
	Density	1.12

	Time (min)	Iteration No.	Node No.	Functional Value
II. Continuous Optimum	.10	230	1	40.0006
First Integer Solution	.24	376	16	59.6186
Optimal Integer Solution	.25	385	17	53.8010
Optimality Proved	.40	499	23	
Time of Search	.40	499	23	

III. Number of Integer Variables Not Integer at Continuous Optimum = 8
Number of Integer Solutions Found = 2
Branches Abandoned While Computing = 12

Table 6.3

Integer Nodes and Functional Values

B-B Node	16	17
Functional Value	59.6186	53.8010
Frequency $w(sec^{-1})$	0.01565	0.01569
Integer Variable	Integer	Integer
N1	1	1
N2	1	1
N3	1	1
N4	1	1
N5	-1	-1
N6	-1	-1
N7	0	0
N8	-1	0

(The B-B Nodes refer to Fig. 6.4)

In order to check the quality of our planar approximations to the objective function $d = z/w$, the test network was re-solved with the optimal value of $w = 0.01569\ sec^{-1}$ ($C = 63.8$ sec), kept fixed. The nonlinearity of the objective function is circumvented in this case. The same integer set resulted, with all other variables being very close to the variable-cycle optimal solution. The functional value value was 53.1272, i.e., a difference of only 1.25%.

6.7 Sensitivity Analysis

Since in practice flows change frequently, the test network was further used to analyse the sensitivity of delay with respect to flows. All flows in the network were changed by the same percentage and the MILPP formulation of Section 5.7 was used to optimize offsets and splits for various cycle times. The results are shown in Fig. 6.5. The decisive role played by the cycle time in reaching optimum operating conditions is illustrated here even more emphatically than before. Again, similarity with single intersection behavior is apparent[26,40]. A clear conclusion is that it certainly makes good sense to try and adapt cycle length to actual traffic conditions. However, how to do this in an optimal manner, taking into account the transient phenomena involved in this process, requires further research.

Table 6.4

Results of Computations

Link No.	Offset sec (cycle)	Green Times sec (cycle)	Delay (LPF)	Delay Due to SDF	Degree of Saturation
101	26.0 (.44)	17.2 (.27)	4.84	0	.674
102	29.3 (.46)	17.2 (.27)	2.23	0	/674
103	38.2 (.60)	24.2 (.38)	12.43	0	.479
104	25.5 (.40)	21.7 (.34)	5.99	.943	.775
105	32.5 (.51)	21.7 (.34)	3.95	0	.535
106	31.2 (.49)	24.2 (.38)	5.74	0	.693
107	21.0 (.33)	30.6 (.48)	3.12	1.660	.868
108	25.5 (.40)	30.0 (.47)	3.00	1.930	.887
109	44.6 (.70)	28.7 (.45)	8.09	1.255	.778
110	19.1 (.30)	37.6 (.59)	2.87	0	.373
111	33.8 (.53)	24.9 (.39)	12.43	2.244	.897
112	30.0 (.47)	28.7 (.45)	4.94	0	.489
113	33.8 (.53)	37.6 (.59)	4.72	0	.565
114	30.6 (.48)	33.1 (.52)	14.60	0	.641
115	24.9 (.39)	30.6 (.48)	10.26	.690	.729
116	24.2 (.38)	24.9 (.39)	4.14	2.244	.897
Input Links					
117		37.6 (.59)	9.43	0	.593
118		33.1 (.52)	12.11	0	.673
119		21.7 (.34)	18.04	0	.535
120		24.2 (.38)	16.12	0	.479
121		23.6 (.37)	18.29	2.714	.901
122		24.9 (.39)	16.06	0	.564
123		30.0 (.47)	15.42	1.930	.887
124		24.2 (.38)	17.14	0	.693

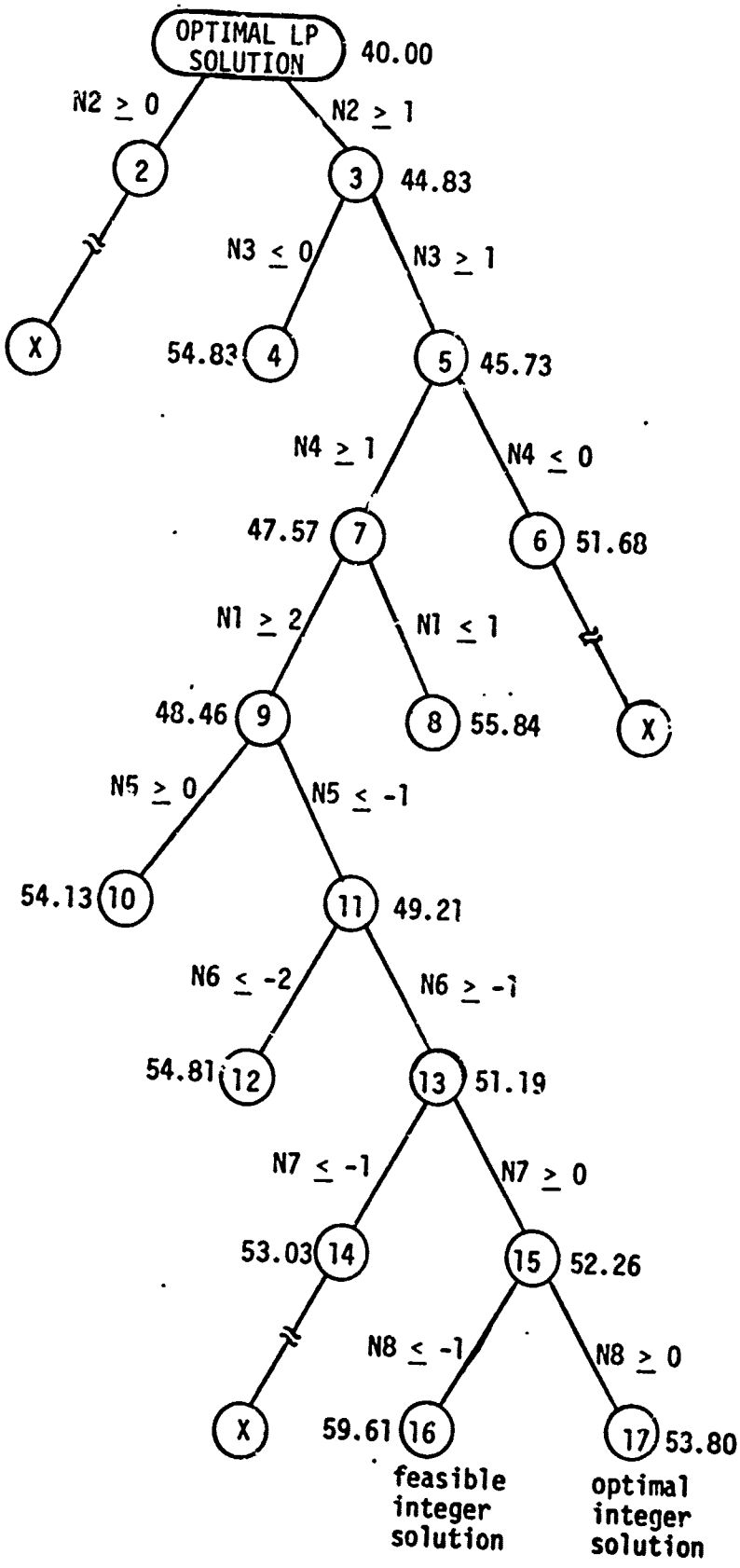


Figure 6.4 Branch and Bound diagram. Offsets, split and cycle time variable.

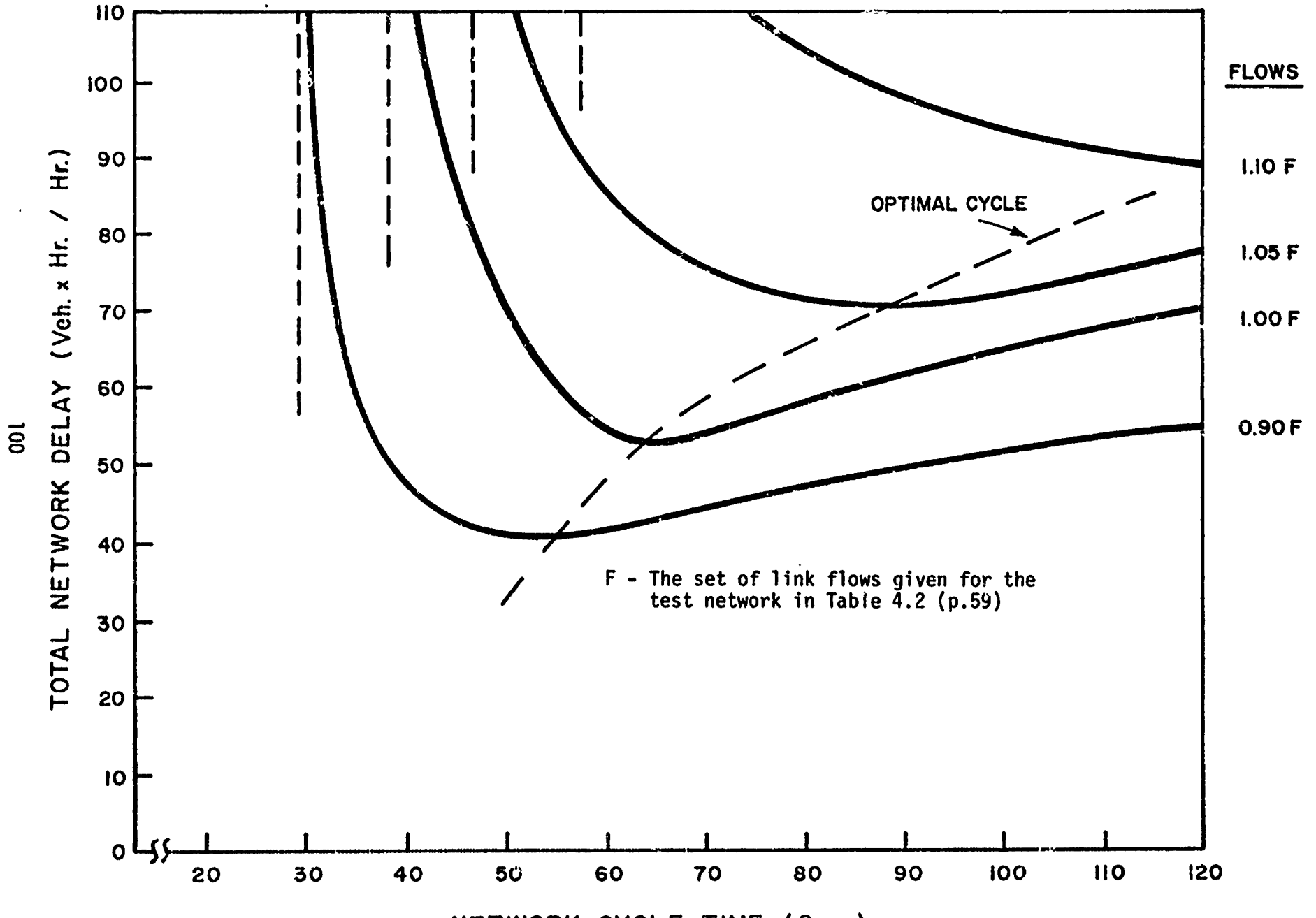


Figure 6.5: Sensitivity of delay and cycle time with respect to flows.

7. Applications and Computational Results of MITROP

This chapter presents computational results that were obtained with the mixed-integer traffic optimization program (MITROP) for the UTCS-1 Test Network, and for an arterial in Waltham, Massachusetts.

These results are considered preliminary since it is the belief of the authors that further iterations with improved data are necessary to obtain results that will, in fact, optimize the performance of the actual networks. In contrast with previous sequential optimization methods, MITROP applies a rigorous optimization procedure for determining all the decision variables in a signal-controlled network--offsets, splits, and cycle time--simultaneously. It is of great importance that the data truly reflect the physical situation to which it is applied. In particular the most heavily loaded intersections should be examined carefully in order to obtain accurate results [35].

7.1 UTCS-1 Test Network

Here we present some computational results for the UTCS-1 Test Network (Fig. 7.1). Every traffic network has its own peculiarities in topology and certain care must be taken in preparing the input data before an optimization procedure is employed. We shall indicate how data is transformed and in particular the handling of extra phases as auxiliary links.

A. Flows

For any given link we are given primary flows secondary flows, and turning movements. First, we add primary and secondary flows $(f_p + f_s)$ to obtain a total flow f_T. This must then be adjusted for turning movements. Given percentages of turning movements, left(%) = a, thru (%) = b, right (%) = c such that a + b + c = 1 we adjust as follows:

$$f_T(1 + 3/4\ a + 1/4c) = F \qquad (7.1)$$

This is again adjusted for trucks and buses, i.e., 3% increase for buses (except on K-street where we use 6%) and a 5% increment for trucks. So the final total flow is 1.08F or 1.11F on K-street.

Platoon lengths are computed as described in Section 3.2.

B. Extra Phases

Additional physical node constraints may occur in a network due to extra phases. For instance, on any link an extra phase, i.e., an additional phase to accommodate turning movement, may be modeled as an auxiliary link running parallel to the main link. This may introduce two additional types of constraints, one reflecting capacity and the other sequencing.

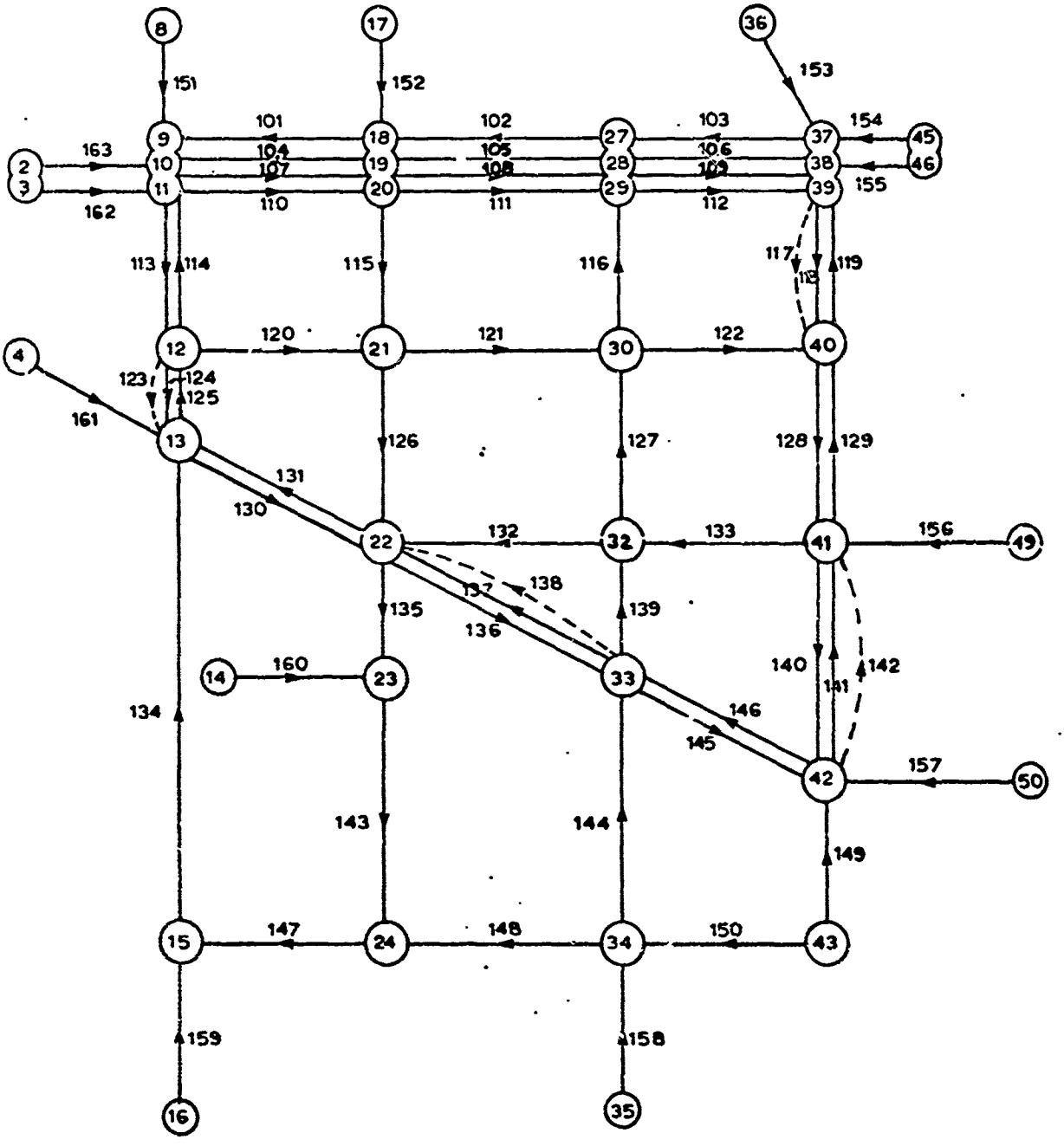


Figure 7.1 UTCS-1 Test Network

If on link i we have a phase for straight thru traffic plus a phase for left turning movements which partially coincides with the main phase in time, an extra link j incident to node n and in the direction of link i may be introduced (Fig. 7.2).

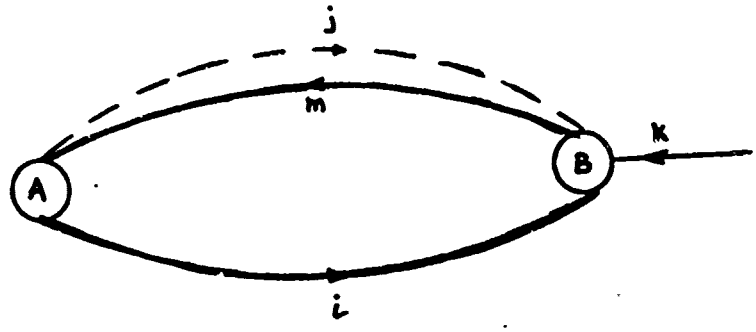


Figure 7.2 Schematic of an extra phase j.

For the duration of green in direction i, we have a green in direction k, which is interrupted by the green for the extra phase. This can be written as

$$g_i + l_i = g_j + l_j + g_k + l_k \tag{7.2}$$

Furthermore, we have that the start of green on the extra phase occurs after the green duration in direction k or

$$\phi_i + g_j + l_j = \phi_k \tag{7.3}$$

This is, in fact, a loop constraint with zero as the integer. Although we have illustrated particular cases, it is in principle the method for handling extra phases or other timing problems by insuring timing, sequencing and by appropriately equating phase duration.

We must also specify input data for an extra phase. Again we are given $f_s + f_p$ and percent of through movement = a and percent turning movement=b such that a + b = 1 on link i.

For link i the total flow is $f_T = a(f_p + f_s)$ and for link j, $f_T = b(f_p + f_s)$, see (Fig. 7.2. Link m is handled as previously indicated. Appropriate adjustments are made for turning movements, then for trucks and buses. We shall have equal platoon lengths on links i and j and also assume that link lengths, lost times, and minimum red times are equal on links i and j.

Since the saturation flow = $\frac{\text{number of lanes}}{\text{headway}}$ we use the following data

Table 7.1

UTCS-1 Test Network: Off Peak

Link Number	Number of Lanes	Headway
117	2	2.4
118	2	2.4
123	1	2.4
124	2	2.4
137	2	2.4
138	1	2.4
141	2	2.4
142	2	2.4

In addition link 113 has a 25.0 sec. minimum red time.

If $f_s > f_p$ on link i (the secondary flow exceeds the primary flow),

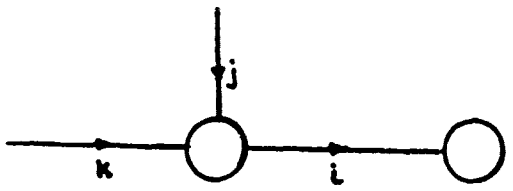


Figure 7.3 Schematic for platoon lengths when $f_s > f_p$.

the platoon length on link i then becomes (see Fig. 7.3),

$$p = g(1 + f_p/f_s)$$

where g is the green time on link j. Further adjustments to this platoon length are as previously indicated. Also offset across link i is measured from the beginning of the green on j rather than from the beginning of green on link k as usual.

Tables 7.2 thru 7.6 represent input data and computational results for the UTCS-1 Test Network: Off Peak.

Table 7.2

Loops of the UTCS-1 Network and Their Corresponding Integer Variables

Loop No.	Integer Variable	Link Numbers				
1	N1	101	110			
2	N2	102	111			
3	N3	103	112			
4	N4	118	119			
5	N5	128	129			
6	N6	140	141			
7	N7	130	131			
8	N8	137	136			
9	N9	145	146			
10	N10	113	114			
11	N11	124	125			
12	N12	139	132	136		
13	N13	115	121	116	105	
14	N14	104	113	115	120	
15	N15	122	119	106	116	
16	N16	125	120	126	131	
17	N17	121	132	127	126	
18	N18	122	128	133	127	
19	N19	141	133	145	139	
20	N20	130	135	143	147	134
21	N21	135	143	144	137	148

Table 7.3

Link Data for UTCS-1 Test Network - Off Peak*

Link No.	Link Length (ft.)	Travel Time (sec.)	Flow (veh/sec)	Sat. Flow (veh/sec)	Platoon Length (Cycle)
101	270	9.82	.101	.417	.511
102	340	12.10	.074	.833	.547
103	430	16.50	.112	.417	.502
104	270	9.82	.158	.417	.511
105	340	12.10	.236	.833	.547
106	430	16.50	.203	.833	.525
107	270	9.82	.176	.833	.668
108	340	12.10	.204	.833	.526
109	430	16.50	.221	.833	.579
110	270	9.82	.125	.417	.501
111	340	12.10	.056	.417	.526
112	430	16.50	.110	.417	.579
113	220	10.71	.163	.417	.413
114	220	11.70	.132	.833	.643
115	220	10.43	.274	.833	.474
116	220	11.65	.253	.833	.601
117	220	9.30	.116	.833	.552
118	220	9.30	.246	.833	.552
119	220	10.11	.265	1.25	.373
120	220	11.39	.103	.833	.640
121	340	15.63	.157	.833	.723
122	430	17.33	.224	.833	.995
123	160	5.68	.013	.417	.640
124	160	5.68	.013	.833	.640
125	160	7.33	.239	.417	.250
126	260	14.06	.213	.833	.601
127	260	12.16	.368	.833	.626
128	260	10.18	.267	.833	.670
129	260	10.35	.279	.833	.640
130	200	10.84	.251	1.25	.447
131	200	11.18	.338	1.25	.986
132	340	12.64	.104	.417	.494
133	430	18.69	.242	.833	.424
134	570	24.57	.172	.833	.765
135	170	6.45	.186	1.66	.669

* See footnote on next page.

Table 7.3 (cont'd.)

Link No.	Link Length (ft.)	Travel Time (sec.)	Flow (veh/sec)	Sat. Flow (veh/sec)	Platoon Length (cycle)
136	380	13.51	.211	1.25	.375
137	380	13.15	.173	.833	.590
138	380	13.15	.061	.417	.590
139	170	7.10	.068	.833	.542
140	370	11.69	.269	.833	.507
141	370	13.51	.156	.833	.580
142	370	13.51	.028	.813	.580
143	310	11.20	.162	1.66	.386
144	310	12.36	.300	.833	.624
145	470	16.67	.302	1.25	.682
146	470	17.53	.206	1.25	.816
147	270	11.70	.068	.833	.405
148	340	14.06	.121	.833	.460
Input Links					
149			.242	.833	
150			.075	.833	
151			.100	.417	
152			.252	.833	
153			.279	1.25	
154			.115	.417	
155			.201	.833	
156			.455	1.25	
157			.315	1.25	
158			.479	.833	
159			.345	.833	
160			.030	.417	
161			.310	1.25	
162			.134	.833	
163			.343	.833	

*(a) Lost time for each link l = 4.0 secs.
(b) Platoon length on input links, p = 1 cycle.
(c) Minimum red time is 15.0 seconds on every link except link 113, where it is 25.0 seconds.

Table 7.4

Results and Statistics for UTCS-1 Test Network - Off Peak

I.	Rows	674
	Columns	996
	Variables	1670
	Integer Variables	21
	Non-zero Elements	2798
	Density	.41

		Time (min)	Iteration No.	Functional Value
II.	Continuous Optimum	.62	604	131.606
	First Integer Solution	1.53	970	164.56
	Second Integer Solution	3.68	1644	164.56

Notice that the second integer solution reached the same value for the objective function as the first, thus the solution was not improved. We set a time limit of four minutes of CPU time. Although the program did not prove optimality under this restriction, it seems that the first integer solution found, although perhaps not optimal is quite near optimal. Other computational experience (see test network in previous sections) tends to support this hypothesis.

Table 7.5

Integer Nodes and Functional Values

UTCS-1 Test Network - Off Peak

Functional Value	164.56	164.56
Frequency $w = (sec)^{-1}$	0.01085	0.01085
Integer Variable	**Integer**	**Integer**
N1	0	0
N2	0	0
N3	0	0
N4	0	0
N5	1	1
N6	0	0
N7	0	0
N8	0	0
N9	0	0
N10	0	0
N11	0	0
N12	-1	-1
N13	-1	-1
N14	-1	-2
N15	-1	-1
N16	-2	-2
N17	-2	-2
N18	0	-1
N19	-2	-2
N20	-2	-2
N21	-1	-1

Table 7.6

Results of Computations

UTCS-1 Test Network - Off Peak

(C = 92 seconds)

Link No.	Offset sec(cycle)	Green Times sec(cycle)	Delay	Delay Due to SDF	Degree of Saturation
101	-1.84 (-.02)	53.5 (.58)	3.59	0	.418
102	-2.76 (-.03)	50.7 (.55)	5.81	0	.162
103	11.05 (.12)	48.8 (.53)	2.67	0	.507
104	-1.84 (-.02)	53.5 (.58)	4.33	0	.653
105	-2.76 (-.03)	50.7 (.55)	8.29	0	.515
106	11.05 (.12)	48.8 (.53)	3.59	0	.460
107	1.84 (.02)	50.7 (.55)	8.66	0	.384
108	2.76 (.03)	48.8 (.53)	6.18	0	.462
109	-11.05 (-.12)	53.5 (.58)	12.44	0	.457
110	1.84 (.02)	50.7 (.55)	2.40	0	.545
111	2.76 (.03)	48.8 (.53)	5.16	0	.253
112	-11.05 (-.12)	53.5 (.58)	12.44	0	.328
113	-18.43 (-.20)	67.3 (.73)	.184	0	.535
114	18.43 (.20)	28.6 (.31)	2.37	0	.511
115	13.82 (.15)	48.8 (.53)	1.57	0	.621
116	33.18 (.36)	34.1 (.37)	10.88	1.172	.821
117	23.04 (.25)	16.6 (.18)	25.99	1.209	.774
118	-15.67 (-.17)	55.3 (.60)	11.71	0	.492
119	15.67 (.17)	28.6 (.31)	2.76	0	.684
120	-7.37 (-.08)	35.0 (.38)	20.46	0	.325
121	57.14 (.62)	24.9 (.27)	20.83	0	.698
122	-20.28 (-.22)	28.6 (.31)	31.61	1.556	.867
123	71.89 (.78)	8.3 (.09)	38.34	0	.346
124	44.24 (.48)	35.9 (.39)	19.45	0	.040
125	7.37 (.08)	67.3 (.73)	0.0	.769	.785
126	43.40 (.46)	35.9 (.39)	14.29	1.704	.656
127	11.06 (.12)	59.0 (.64)	0.0	0	.690
128	48.85 (.53)	33.2 (.36)	22.95	1.685	.890
129	43.32 (.47)	34.1 (.37)	18.53	1.663	.905
130	21.20 (.23)	30.4 (.33)	5.16	0	.608
131	-21.20 (-.23)	47.9 (.52)	21.29	0	.520
132	-16.59 (-.18)	57.1 (.62)	13.09	0	.402
133	20.28 (.22)	36.9 (.40)	.92	1.366	.725
134	47.93 (.52)	23.0 (.25)	25.25	0	.826
135	0.0 (0.0)	61.8 (.67)	.92	0	.167

Table 7.6 (cont'd)

Link No.	Offset sec(cycle)	Green Times sec(cycle)	Delay	Delay Due to SDF	Degree of Saturation
136	7.37 (.08)	38.7 (.42)	3.59	0	.402
137	-7.37 (-.08)	58.1 (.63)	6.36	0	.330
138	28.57 (.31)	22.1 (.24)	21.57	0	.610
139	7.37 (.08)	47.0 (.51)	3.59	0	.631
140	14.75 (.16)	44.2 (.48)	1.29	0	.723
141	-14.75 (-.16)	45.2 (.49)	18.34	0	.382
142	23.04 (.25)	7.4 (.08)	33.18	0	.431
143	6.45 (.07)	39.6 (.43)	0.0	0	.227
144	23.96 (.26)	46.1 (.50)	5.90	0	.720
145	-.92 (-.01)	40.6 (.44)	18.16	0	.549
146	.92 (.01)	38.7 (.42)	20.09	0	.389
147	32.26 (.35)	16.6 (.18)	10.23	0	.454
148	13.82 (.15)	44.2 (.48)	0.0	0	.303
Input Links					
149		44.2 (.48)	19.26	0	.605
150		12.9 (.14)	36.31	0	.643
151		28.6 (.31)	28.94	1.453	.774
152		31.3 (.34)	28.02	1.667	.890
153		28.6 (.31)	28.29	0	.720
154		53.5 (.58)	12.63	0	.475
155		53.5 (.58)	12.07	0	.416
156		39.6 (.43)	23.87	1.146	.827
157		40.6 (.44)	21.20	0	.573
158		71.0 (.77)	6.36	.084	.747
159		71.0 (.77)	3.13	0	.539
160		22.1 (.24)	29.68	0	.300
161		47.9 (.52)	15.48	0	.477
162		53.5 (.58)	10.23	0	.277
163		53.5 (.58)	14.93	0	.710

7.2 Waltham Arterial

An example of an arterial is given which contains 11 intersections on Main Street in Waltham, Massachusetts, Fig. 7.4. Tables 7.7 and 7.8 represent input data while Tables 7.9, 7.10, and 7.11 give results. Space-time diagrams of the results are given in Fig. 7.5. Part (a) of the figure represents westbound flows, part (b) eastbound flows, the difference being caused by multiple phasing at some intersections. Only the pertinent phases in a given direction are considered. Bandwith, a geometrical quantity, is not a criterion in our optimization procedure. The settings provided by MITROP minimize total delay by taking explicit account of all traffic movements affected by the arterial signals.

Table 7.7

Loops for the Arterial and their Corresponding Integer Variables

Loop No.	Integer Variable	Link Numbers	
1	N1	104	107
2	N2	108	109
3	N3	110	111
4	N4	112	114
5	N5	115	116
6	N6	117	118
7	N7	119	120
8	N8	121	122
9	N9	123	125
10	N10	101	103

Table 7.8

Link Data for the Arterial*

Link No.	Link Length (ft)	Travel Time (sec)	Flow (veh/sec)	Sat. Flow (veh/sec)	Platoon Length (cycle)
101	810	23.82	.106	.534	.499
102	810	23.82	.204	1.111	.499
103	810	23.82	.270	1.467	.612
104	1105	32.50	.182	.979	.713
105	1105	32.50	.052	.445	.713
106	1105	32.50	.047	.445	.543
107	1105	32.50	.175	.978	.543
108	755	22.20	.223	.890	.840
109	755	22.20	1222	.890	.798
110	800	23.50	.201	.890	.678
111	800	23.50	.228	1.328	.848
112	360	10.60	.152	.801	.388
113	360	10.60	.089	.534	.388
114	360	10.60	.287	.979	.600
115	230	6.76	.162	1.156	.388
116	230	6.76	.336	1.332	.663
117	830	24.41	.233	1.111	.855
118	830	24.41	.233	1.111	.755
119	410	12.06	.208	1.111	.588
120	410	12.06	.230	1.022	.750
121	340	10.00	.298	.979	.463
122	340	10.06	.296	1.068	.588
123	1050	30.88	.224	1.068	.915
124	1050	30.88	.083	.498	.564
125	1050	30.88	.193	.979	.564
Input Links					
126			.325	1.289	
127			.039	.445	
128			.138	.623	
129			.096	.712	
130			.037	.667	
131			.011	.534	
132			.106	.979	
133			.072	.489	
134			.125	.489	
135			.189	1.156	
136			.030	.890	
137			.088	.623	
138			.176	.979	
139			.159	1.088	
140			.086	1.200	
141			.089	.489	
142			.076	.489	

*(a) Lost time for each link, l = 4.0 secs.
(b) Platoon length on input links, p = 1 cycle.
(c) Minimum red time on each link is 15 secs.
(d) Velocity on each link is 34 ft/sec.

Table 7.9

Results and Statistics for the Arterial

I.	Rows	427
	Columns	206
	Variables	631
	Integer Variables	10
	Non-zero Elements	1725
	Density	0.64

		Time (min.)	Iteration No.	Functional Value
II.	Continuous Optimum	.25	410	79.6803
	First Integer Solution	.53	571	91.8803
	Optimal Integer Solution	.53	571	91.8803
	Optimality Proved	.90	743	
	Time of Search	.90	743	

III.	Number of Integer Variables not at Continuous Optimum	10
	Number of Integer Solutions Found	1

Table 7.10

Integer Nodes and Functional Values for the Arterial

Functional Value	91.8803
Frequency w(sec $^{-1}$)	.01451
Integer Variable	Integer
N1	1
N2	1
N3	1
N4	0
N5	0
N6	1
N7	0
N8	0
N9	1
N10	2

Table 7.11

Results and Computations for the Waltham Arterial

C = 69 seconds

Link No.	Offset sec (cycle)	Green Times sec (cycle)	Delay	Delay due to SDF	Degree of Saturation
101	37.26 (.54)	33.12 (.48)	6.60		.414
102	37.26 (.54)	21.39 (.31)	6.60		.592
103	44.16 (.64)	22.08 (.32)	10.05		.575
104	46.92 (.68)	22.08 (.32)	14.90		.581
105	73.14 (1.06)	14.49 (.21)	20.32		.347
106	56.58 (.82)	13.80 (.20)	11.84		.528
107	22.08 (.32)	48.30 (.70)	0.0		.255
108	49.68 (.72)	30.36 (.44)	13.84		.569
109	19.32 (.28)	44.16 (.64)	6.20		.390
110	26.22 (.38)	44.16 (.64)	1.40		.353
111	42.78 (.62)	19.32 (.28)	19.89		.613
112	12.42 (.18)	19.32 (.28)	11.26		.678
113	35.88 (.52)	15.87 (.23)	17.34	1.343	.725
114	-12.42 (-.18)	30.36 (.44)	8.33		.666
115	19.32 (.28)	14.49 (.21)	6.24		.667
116	-19.32 (-.28)	48.99 (.71)	4.68		.355
117	46.23 (.67)	34.50 (.50)	12.39		.401
118	22.77 (.33)	53.82 (.78)	0.0		.269
119	- 1.38 (-.02)	53.82 (.78)	0.0		.240
120	1.38 (.02)	32.43 (.47)	12.19		.479
121	9.66 (.14)	32.43 (.47)	0.0		.648
122	- 9.66 (-.14)	29.67 (.43)	18.68		.645
123	48.30 (.70)	29.67 (.43)	16.72		.488
124	50.37 (.73)	19.32 (.28)	9.86		.595
125	20.70 (.30)	48.99 (.71)	0.0		.278
Inputs Links					
126		18.63 (.37)	18.77		.681
127		12.42 (.18)	25.60		.487
128		17.25 (.25)	24.10	2.423	.886
129		14.49 (.21)	24.36		.642
130		14.49 (.21)	23.35		.264
131		6.9 (.10)	28.13		.206
132		11.73 (.17)	26.19		.637
133		17.25 (.25)	23.06		.589
134		25.53 (.37)	18.90	.480	.691
135		16.56 (.24)	23.79		.681
136		17.25 (.25)	21.56		.135
137		13.11 (.19)	25.41		.574
138		20.01 (.29)	21.51		.620
139		53.13 (.77)	2.40		.193
140		8.28 (.12)	28.14		.597
141		27.6 (.40)	16.40		.455
142		35.19 (.51)	10.93		.305

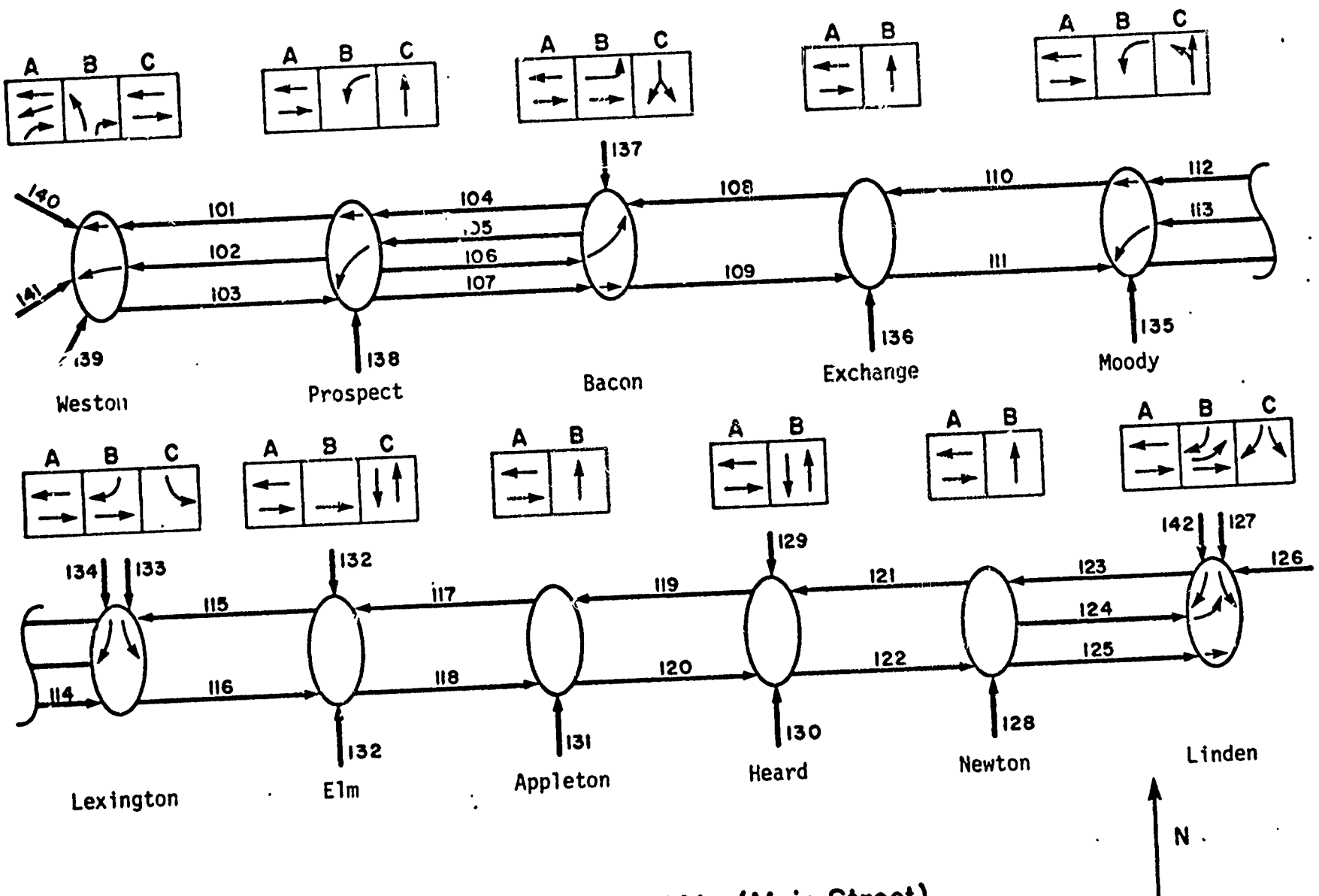


Figure 7.4: WALTHAM ARTERIAL (Main Street)

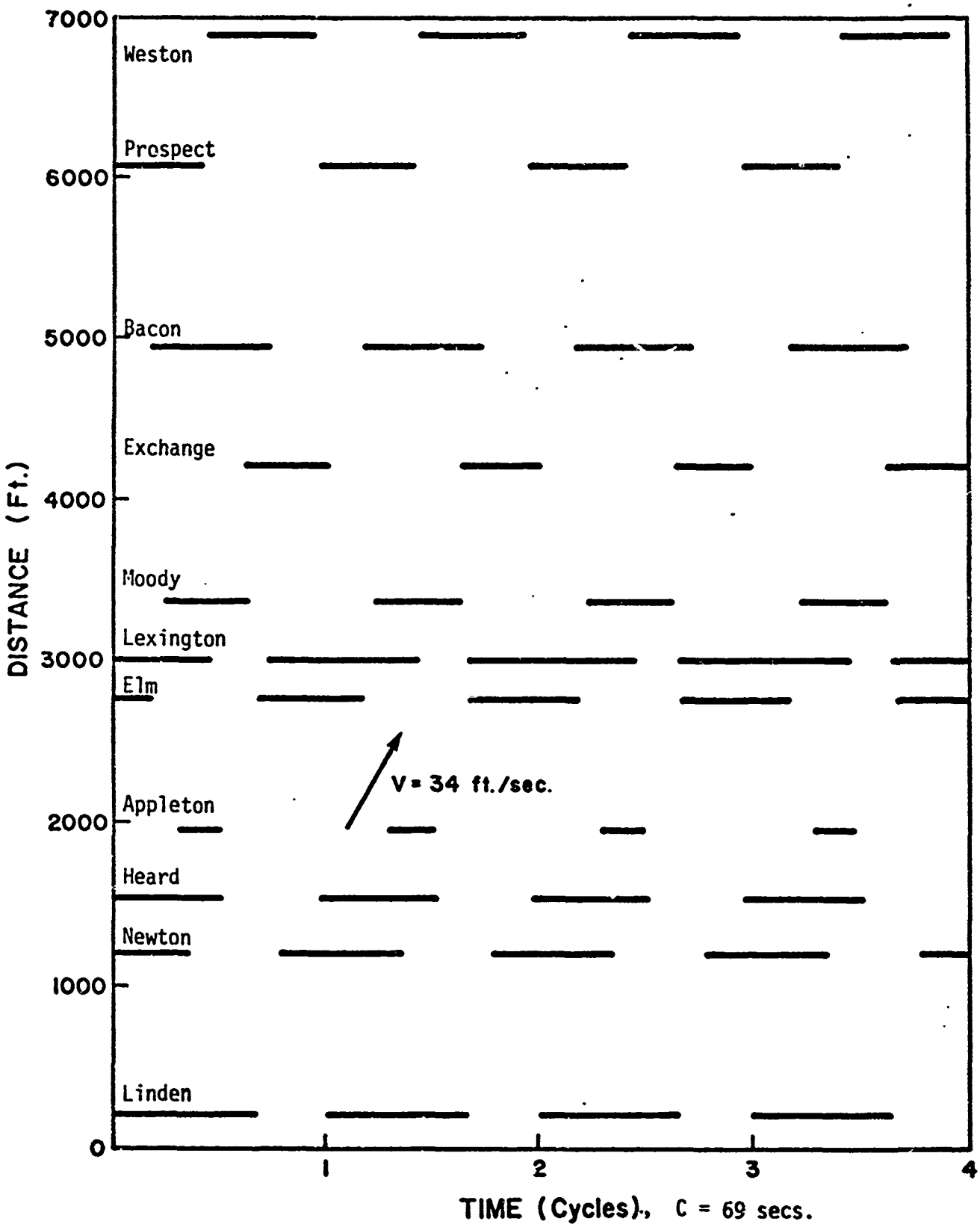


Figure 7.5(a): Time-Space Diagram for Waltham Arterial, Westbound

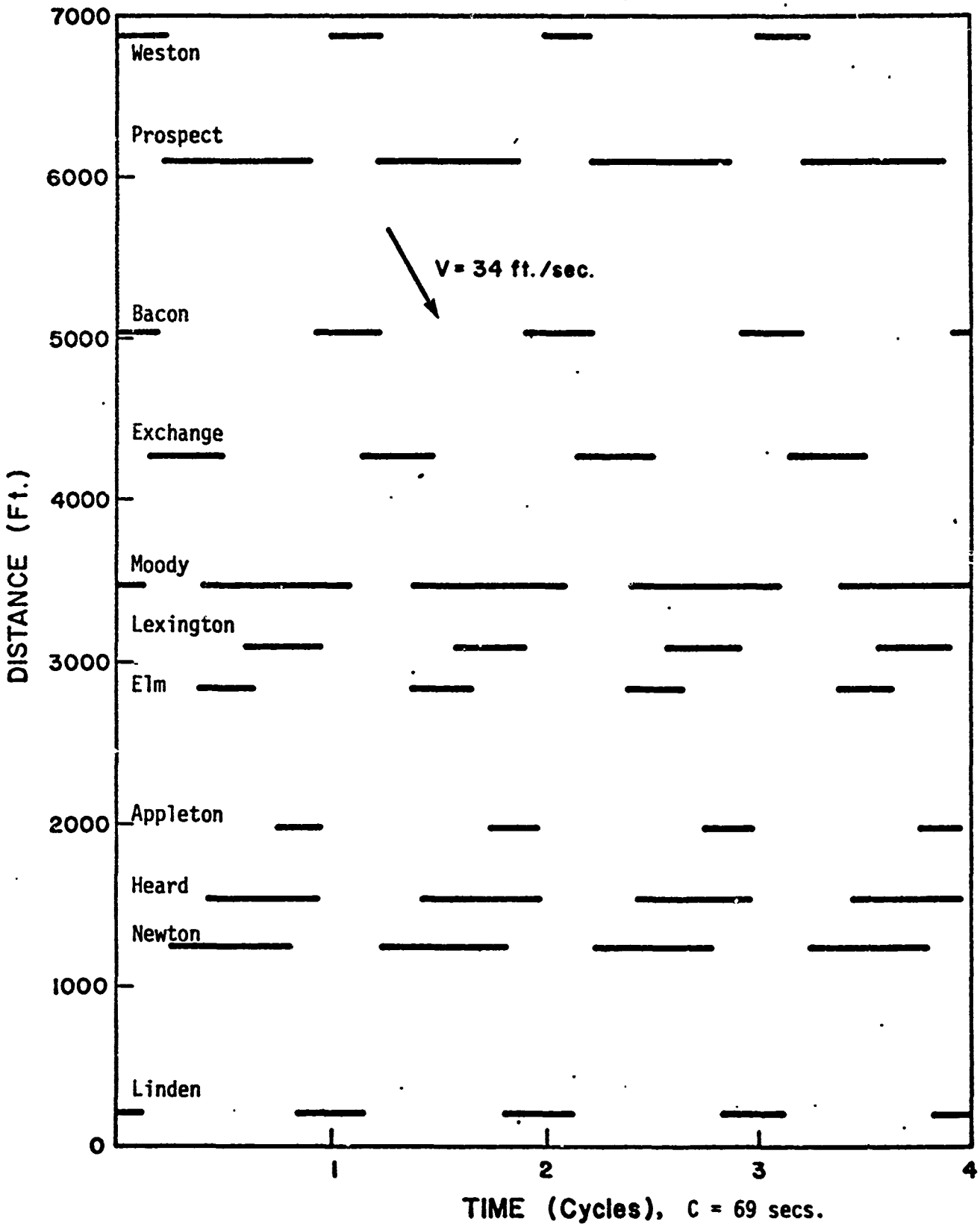


Figure 7.5(b): Time-Space Diagram for Waltham Arterial, Eastbound.

8. DISCUSSION OF RESULTS

The preceding chapters described the model formulation of the traffic signal network timing problem and the computational aspects of applying mixed-integer linear programming techniques (MILP) for its optimization. The techniques were used to calculate offsets (Chapter 4) offsets and splits (Chapter 5), and offsets, splits and cycle time simultaneously (Chapter 6).

The conventional procedure for determining the signal control variables is to use a sequential decision process: A common cycle time is selected for the network first. Then the splits at each intersection are determined according to the proportions of demand/capacity ratios on conflicting approaches. Finally, linking of the signals is achieved by an appropriate method for selection of a fundamental set of offsets throughout the network.

Experience of researchers and practitioners in the urban traffic control field has shown that cycle time may well be the most important control variable for determining signal settings in a network [38]. Current approaches for selecting a cycle time can be divided into two classes. The first class is the node approach. Since through capacity increases with cycle length, this approach is based on analyzing the capacity requirements of each intersection in the network. The common cycle time is determined according to the requirements of the most heavily loaded intersection, i.e., the intersection with the highest sum of demand/capacity ratios on conflicting signal phases. A procedure that is used for a single intersection, such as Webster's method [40], is then used to calculate the cycle length. This approach has been primarily used in conjunction with offset optimization methods such as Combination and TRANSYT [17]. The main deficiency in this approach is that the interaction of flows in the spatial road network structure of the area is disregarded. A formula devised for an isolated intersection, assuming randomly distributed arrival times of cars, is not necessarily valid in a network situation where flows are fed from adjacent intersections. The result is generally a cycle time that is too long, thus causing excessive delays.

The second class is the network approach. In this case an attempt is made to select a cycle time that, while satisfying the capacity requirements at each intersection, is also congruent with the particular network structure at hand. Simple examples in this category are the arterial progression schemes in which a cycle that produces maximal bandwidths is selected according to distance and speed data (e.g., [3,24,25]). The underlying principle is that the optimal progression for given block-length pattern and desired speeds is strongly dependent on cycle time. In a general network this approach is principally used by SIGOP [37]. A predetermined number of cycle times are scanned in this method. For each cycle, offsets are optimized by the OPTMIZ subroutine and performance is evaluated by a coarse simulation of traffic flow through the network (the VALUAT subroutine). The optimal set of cycle and offsets is selected according to the results obtained by VALUAT. TRANSYT indicates also the possibility to iterate on cycle time in conjunction with its hill-climbing procedure for offset selection [34]. However, computational considerations seem to rule out this possibility in practice. Two deficiencies of the network approach in SIGOP are apparent: first, the offset optimization procedure determines a local optimum rather than global optimum and, second, stochastic effects on link

performance are ignored. These effects do not affect the selection of offsets at a fixed cycle time, but are of prime importance in evaluating a range of cycle times. They become pronounced as a signalized intersection approaches its capacity and in an optimal procedure would deter the cycle time from assuming values close to the minimum.

This study models both the deterministic and stochastic effects, via the link performance function (Chapter 3) and the saturation deterrence function (Section 5.5), respectively. For the first time a method is presented that optimizes all the decision variables--cycle time, splits, and offsets--simultaneously. The results obtained with this approach show promise of substantial improvements in network performance when compared to conventional procedures. The optimal settings for the test network in Chapter 6, when all the decision variables were considered simultaneously (objective function = 53.8010), showed an improvement of 11.6% in network performance with respect to the settings obtained in Chapter 4, when the decision variables were considered sequentially (objective function = 60.8741). The interplay between the LPF and the SDF in the objective function for variable cycle time (for the test network analyzed in the report) is shown in Fig. 6.1 and is of fundamental importance in analyzing the performance of area traffic control systems. Furthermore, the results in Fig. 6.5 show that the timings, and in particular the cycle time, are sensitive to flows and substantial gains can be achieved by setting the system in accordance with the actual traffic conditions. From a computational standpoint the method is not at the present time implementable as an online real-time procedure. Nevertheless, MITROP presents an innovative, comprehensive and systematic approach to traffic signal settings in a network and should be further developed in several directions.

First, on an offline basis, the method should be developed into an operational tool for use by practicing traffic engineers. Results obtained in this study demonstrate that sizeable networks can be optimized in relatively short times, on the order of a few minutes. Hence, the method is capable of being developed into an interactive computer program which can be put on a time-sharing network and made easily accessible.

Second, the method should be applied to several urban networks to test its calibration procedures and evaluate its performance on the street.

Third, the insight and knowledge gained in this research can be used for modeling the traffic process in its transient behavior and developing the analytical tools for its online optimization, which is indeed the third-generation goal. However, there still seems to be a long way before urban traffic control and management can be turned into an automatic feedback control system. The traffic analyst still plays an important role in the loop going from traffic data to signal settings. Therefore, first- and second-generation strategies will seemingly continue to constitute the backbone of traffic control procedures for a long time to come.

9. REFERENCES

1. ALLSOP, R.E. (1968a). Selection of Offsets to Minimize Delay to Traffic in a Network Controlled by Fixed-Time Signals. Transportation Science 2, 1-13.

2. ALLSOP, R.E. (1968b). An Analysis of Delays to Vehicle Platoons at Traffic Signals. Proc. Fourth Intern. Symp. on the Theory of Traffic Flow, University of Karlsruhe, Germany.

3. BAERWALD, J.E., Editor (1965). Traffic Engineering Handbook, Third Edition. Institute of Traffic Engineers, Washington, D.C.

4. CHUNG, C.C. and N. GARTNER (1973). Acceleration Noise as a Measure of Effectiveness in the Operation of Traffic Control Systems. Working Paper OR 015-73, Operations Research Center, M.I.T., Cambridge, Mass.

5. CLAYTON, A.J.H. (1940-41). Road Traffic Calculations. J. Inst. of Civil Engineers, Vol. 16.

6. EISNER, M.J. (1972). Synchronizing Traffic Signals in a Saturated Network: A Deterministic Optimization Model. Technical Report No. 155, Dept. of Operations Research, Cornell University, Ithaca, NY.

7. GARFINKEL, R.S. and G.L. NEMHAUSER (1972). Integer Programming. John Wiley, New York.

8. GARTNER, N. (1972a). Optimal Synchronization of Traffic Signal Networks by Dynamic Programming. Traffic Flow and Transportation (G.F. Newell, Editor), American Elsevier Publishing Comp., New York, 281-295.

9. GARTNER, N. (1972b). Constraining Relations Among Offsets in Synchronized Signal Networks. Transportation Science 6, 88-93.

10. GARTNER, N. (1973). Microscopic Analysis of Traffic Flow Patterns for Minimizing Delay on Signal Controlled Links. Highway Research Record No. 445, 12-23.

11. GARTNER, N. and J.D.C. LITTLE (1973). The Generalized Combination Method for Area Traffic Control. Working Paper OR 020-73, Operations Research Center, M.I.T., Cambridge, Mass.

12. GAZIS, D.C. (1965). Traffic Control, Time-Space Diagrams, and Networks. Traffic Control-Theory and Instrumentation (T.R. Horton, Editor), Plenum Press, N.Y., 47-63.

13. GRAHAM, E.F. and D.C. CHENU (1962). A Study of Unrestricted Platoon Movement of Traffic. Traffic Engineering 32, 11-13.

14. HEWTON, J.T. (1968). The Metropolitan Toronto Traffic Control Signal System. Proc. IEEE 56, 577-599.

15. HILLIER, J.A. (1966). Appendix to Glasgow's Experiment in Area Traffic Control. Traffic Engineering and Control 7, 569-571.

16. HILLIER, J.A. and R. ROTHERY (1967). The Synchronizaiton of Traffic Signals for Minimum Delay. Transportation Science 1, 81-94.

17. HOLROYD, J. (1972). The Practical Implementation of Combination Method and TRANSYT Programs. Colloquium on Area Traffic Control, IEE (GB).

18. HOLROYD, J. and J.A. HILLIER (1971). The Glasgow Experiment: Plident and after. Road Research Laboratory Report LR 384, Crowthorne, Berkshire, U.K.

19. HUDDART, K.W. (1969). The Importance of Stops in Traffic Signal Progressions. Transportation Research, Vol. 3, pp. 143-150.

20. HUDDART, K.W. and E.D. TURNER (1969). Traffic Signal Progressions-G.L.C. Combination Method. Traffic Engineering & Control, November 1969, 320-322.

21. IBM Corporation (1971). Mathematical Programming System Extended (MPSX) --Mixed Integer Programming (MIP) Program Description. Program Number 5734-XM4. Technical Publications Department, White Plains, New York.

22. KAPLAN, J.A. and L.D. POWERS (1973). Results of SIGOP-TRANSYT Comparison Studies. Traffic Engineering 43 (12).

23. KORSAK, A. (1973). An Algorithm for Globally-Optimal Nonlinear-Cost Multidimensional Flows in Networks and Some Special Applications. Operations Research 21, 225-239.

24. LITTLE, J.D.C. (1966). The Synchronization of Traffic Signals by Mixed-Integer Linear Programming. Operations Research 14, 568-594.

25. MATSON, T.M., SMITH, W.S. and F.W. HURD (1955). Traffic Engineering. McGraw-Hill, New York.

26. MILLER, A.J. (1963). Settings for Fixed-Cycle Traffic Signals. Operational Research Quarterly 14, 373-386.

27. MILLER, A.J. (1968). The Capacity of Signalized Intersections in Australia. Bulletin No. 3, Australian Road Research Board.

28. MORGAN, J.T. and J.D.C. LITTLE (1964). Synchronizing Traffic Signals for Maximal Bandwidth. Operations Research 12, 896-912.

29. MORSE, P.M. (1958). Queues, Inventories and Maintenance. John Wiley, N.Y

30. NEMETH,Z.A. and R.L. VECELLIO (1970). Investigation of the Dynamics of Platoon Dispersion. Highway Research Record No. 334, 23-33.

31. NEWELL, G.F. (1964). Synchronizaiton of Traffic Lights for High Flow. Quarterly of Applied Mathematics 21, 315-324.

32. NEWELL, G.F. (1965). Approximation Methods for Queues with Application to the Fixed-Cycle Traffic Light. SIAM Review 7, 223-240.

33. NEWELL, G.F. (1968). Traffic Signal Synchronization for High Flows on a Two-Way Street. Proc. Fourth Intern Symp. on the Theory of traffic flow, University of Karlsruhe, Germany.

34. ROBERTSON, D.I. (1969). TRANSYT: A Traffic Network Study Tool. Road Research Laboratory Report LR 253, Crowthorne, Berkshire, U.K.

35. ROBERTSON, D.I. (1974). Presentation at Workshop on Intersection Capacity, 53rd Annual Meeting of the Highway Research Board, Washington, D.C.

36. SCHLAEFLI, J.L. (1972). Computerized Traffic Signal Systems: A Future. Traffic Engineering 42.

37. Traffic Research Corporation (1966). SIGOP: Traffic Signal Optimization Program, A Computer Program to Calculate Optimum Coordination in a Grid Network of Synchronized Traffic Signals. PB 173 738.

38. WAGNER, F.A., F.C. BARNES, and D.R. GERLOUGH (1971). Improved Criteria for Traffic Signal Systems in Urban Networks. NCHRP Report 124, Highway Research Board, Washington, D.C.

39. WARDROP, J.G. (1952). Some Theoretical Aspects of Road Traffic Research. Proc. Inst. Civil Eng., Part II, 325-378.

40. WEBSTER, F.V. (1958). Traffic Signal Settings. Road Research Technical Paper No. 39, Her Majesty's Stationery Office, London.

41. WHITING, P.D. (1972). The Glasgow Traffic Control Experiment: Interim Report on SIGOP and TRANSYT. Road Research Laboratory Report LR 430, Crowthorne, Berkshire, U.K.

42. WIEDEMANN, R. (1968). Verkehrsablauf hinter Lichtsignalanlagen. Strassenbau und Strassenverkehrstechnik, Heft 74, Bundesminister für Verkehr, Bonn, Germany.

43. WORMLEIGHTON, R. (1965). Queues at a Fixed Time Traffic Signal with Periodic Random Input. Canadian Operations Research Society (CORS) Journal 3, 129-141.